

Aalto University  
School of Science  
Master's Programme in Computer, Communication and Information Sciences

Xi Hu

# **Motion Palette:**

## **Motion Exploration Support Tool for Dynamic Visual Effects**

Master's Thesis  
Espoo, November 25, 2019

Supervisor: Professor Antti Oulasvirta, Aalto University  
Professor Wendy Mackay, Paris-Sud University  
Advisor: Janin Koch  
Stacy Hsueh  
Baptiste Caramiaux

Aalto University

School of Science

 Master's Programme in Computer, Communication and  
 Information Sciences

 ABSTRACT OF  
 MASTER'S THESIS

<b>Author:</b>	Xi Hu		
<b>Title:</b>	Motion Palette: Motion Exploration Support Tool for Dynamic Visual Effects		
<b>Date:</b>	November 25, 2019	<b>Pages:</b>	61
<b>Major:</b>	Human-Computer Interaction and Design	<b>Code:</b>	SCI3020
<b>Supervisor:</b>	Professor Antti Oulasvirta Professor Wendy Mackay		
<b>Advisor:</b>	Janin Koch Stacy Hsueh Baptiste Caramiaux		
<p>The parametric space exploration tools for motion designers and visual artists are still painful to use, so are the similar tools for other parametric designers and generative artists. The disparity between their tools and creation causes trivial trial and error process to find the right parameter combination. Besides, little customization support is provided to narrow and expand their parametric space during the exploration. My goal is to improve existing tools or create new ones to better support motion design process. In this paper, I present a new creativity support tool, Motion Palette, to facilitate motion exploration for visual artists specialized in dynamic visual effects design. Based on interviews of creative process in the first user study, I designed the tool and experimented with potential users in a second study. Results show Motion Palette works best with traditional exploration tools, e.g. sliders, outrating itself or sliders alone.</p>			
<b>Keywords:</b>	motion design, parametric space, exploration, customization, creativity support tool		
<b>Language:</b>	English		

# Acknowledgements

I would like to thank my supervisor from Paris-Sud University, Wendy Mackay, for her trust and support in giving me freedom to choose the topic I like, and guiding me to the right directions when I feel lost.

I would like to thank Stacy Hsueh and Baptiste Caramiaux as well, who act as my advisors from University of Paris-Sud. Stacy followed my whole research all along and provided me with a lot of precious suggestions. Baptiste pointed to me new possibilities that I may further pursue in future studies.

I also sincerely thank Antti Oulasvirta and Janin Koch, my supervisor and advisor from Aalto University, for their trust, encouragement and suggestions for my research.

Besides, I feel grateful for my colleagues and all the participants who have joined me for the studies.

In the end, I want to thank my friends and family for their loving support in getting me through a hard time. Without their backup, I would never have completed this research.

Espoo, November 25, 2019

Xi Hu

# Contents

<b>1</b>	<b>Introduction</b>	<b>7</b>
1.1	Problem Statement . . . . .	7
1.1.1	Disparity Between Tools and Creations . . . . .	7
1.1.2	Little Support for Exploration . . . . .	7
1.2	Research Question . . . . .	7
1.3	Contributions . . . . .	8
1.3.1	Artifact Contributions . . . . .	8
1.3.2	Empirical Contributions . . . . .	8
1.3.3	Theoretical Contributions . . . . .	9
1.4	Structure of the Thesis . . . . .	9
<b>2</b>	<b>Background</b>	<b>10</b>
2.1	Research Solving the Problems . . . . .	10
2.1.1	Disparity Between Tools and Creations . . . . .	10
2.1.1.1	Running Simulation in Real Time . . . . .	10
2.1.1.2	Enhancing Node Based Visual Programming Tools . . . . .	10
2.1.1.3	Reducing Manual Manipulation of Parameters	11
2.1.2	Little Support for Exploration . . . . .	12
2.2	Research Inspiring My Solution . . . . .	12
2.2.1	Instrumental Interaction and Substrates . . . . .	12
2.2.1.1	Information Substrates . . . . .	12
2.2.1.2	Instrumental Interaction . . . . .	13
2.2.2	Parametric Space Exploration Tools Applying Instru- mental Interaction and Substrates . . . . .	13
2.2.2.1	Customized Color Space Exploration . . . . .	13
2.2.2.2	Engineering Parametric Space Exploration . .	13
2.2.2.3	Character Facial Space Exploration . . . . .	14



<b>3</b>	<b>Methods</b>	<b>15</b>
3.1	Critical Object Interview . . . . .	15
3.2	Thematic Analysis . . . . .	16
3.3	Experimental Design . . . . .	16
3.3.1	Within-Subject / Between-Subject Design . . . . .	16
3.3.2	Quantitative / Qualitative Measures . . . . .	16
3.4	Structured Observation . . . . .	17
<b>4</b>	<b>Study One</b>	<b>18</b>
4.1	Interview Study . . . . .	18
4.1.1	Participants . . . . .	18
4.1.2	Procedure . . . . .	18
4.1.3	Data Collection . . . . .	18
4.1.4	Analysis . . . . .	19
4.2	Results and Discussion . . . . .	19
4.2.1	Thematic Analysis . . . . .	19
4.2.1.1	Constraints: Trial and Error . . . . .	19
4.2.1.2	References: Extract and Combine, Adapt and Repurpose . . . . .	20
4.2.1.3	Presetting, Template and Filter: Reuse . . . . .	22
4.2.1.4	Procedural Operations: Struggle to Cross Learning Gap . . . . .	23
4.2.1.5	Tool Stack: Render in 3D and Polish in 2D . . . . .	24
4.2.2	Design Implications . . . . .	24
4.2.2.1	Constraints: Trial and Error . . . . .	24
4.2.2.2	References: Extract and Combine, Adapt and Repurpose . . . . .	25
4.2.2.3	Presetting, Template and Filter: Reuse . . . . .	25
4.2.2.4	Procedural Operations: Struggle to Cross Learning Gap . . . . .	25
<b>5</b>	<b>Study Two</b>	<b>27</b>
5.1	Design Process . . . . .	27
5.1.1	Brainstorming . . . . .	27
5.1.1.1	Direct Control . . . . .	27
5.1.1.2	Stylistic Toolbox . . . . .	27
5.1.1.3	Customized Exploration . . . . .	28
5.1.2	Converging Ideas . . . . .	28
5.1.3	Use Scenario . . . . .	28
5.1.4	Design Concept . . . . .	28
5.1.5	Prototyping . . . . .	31

5.1.6	Implementation . . . . .	31
5.1.7	Features . . . . .	34
5.1.7.1	A Gallery of Saved States . . . . .	34
5.1.7.2	State Copies that Form Palettes . . . . .	34
5.1.7.3	Palette Representation . . . . .	35
5.1.7.4	Palette Snapshots in Motion Scheme Panel . . . . .	36
5.2	Quasi-Experiment: Structured Observation . . . . .	38
5.2.1	Experimental Design . . . . .	38
5.2.1.1	Task I: Reaching a Target Effect . . . . .	38
5.2.1.2	Task II: Free Exploration . . . . .	39
5.2.2	Participants . . . . .	40
5.2.3	Apparatus . . . . .	40
5.2.4	Procedure . . . . .	41
5.2.5	Data Collection . . . . .	41
5.2.6	Analysis . . . . .	42
5.3	Objective Results . . . . .	42
5.3.1	Task I: Reaching a Target Effect . . . . .	42
5.3.1.1	Completion Time (only for reference) . . . . .	42
5.3.1.2	Number of Resets and Trials . . . . .	43
5.3.1.3	Distance to the Target . . . . .	43
5.3.2	Task II: Free Exploration . . . . .	45
5.3.2.1	Completion Time (only for reference) . . . . .	45
5.4	Subjective Results . . . . .	46
5.4.1	Self-Reported Measures (Quantitative) . . . . .	46
5.4.1.1	Task I: Reaching a Target Effect . . . . .	47
5.4.1.2	Task II: Free Exploration . . . . .	48
5.4.2	Debriefing on Strategies and Preferences (Qualitative) . . . . .	51
5.4.2.1	Task I: Reaching a Target Effect . . . . .	51
5.4.2.2	Task II: Free Exploration . . . . .	52
5.5	Discussion . . . . .	54
5.5.1	New Interpolation Strategy . . . . .	54
5.5.2	Natural Interaction with Palettes . . . . .	55
5.5.3	Bidirectional Control between Palettes and Sliders . . . . .	55
5.5.4	Machine Learning User Preferences . . . . .	55
<b>6</b>	<b>Conclusions</b>	<b>56</b>
6.1	Conclusion . . . . .	56
6.1.1	Uncover Exploratory Needs for Motion Designers . . . . .	56
6.1.2	Design and Evaluate Motion Palette . . . . .	56
6.2	Future Work . . . . .	57

# Chapter 1

## Introduction

### 1.1 Problem Statement

#### 1.1.1 Disparity Between Tools and Creations

In parametric design and generative art, designers and artists often fall into struggling with their tools rather than focusing on the artistic expression itself [28]. Unlike traditional artists, e.g. painters, directly manipulating visual elements, their creations are controlled by parameters and dynamically generated, thus influencing their creative process to be confined in text based programming, node based visual programming, or parameters manipulation in properties panels. This disparity between the tools in their creative process and their final creations leads to fumbling back and forth in between and searching for the right parameters that control the right portion of the final work [28].

#### 1.1.2 Little Support for Exploration

The creative process usually involves iterations of diverging and converging ideas. Same applies to parametric design and generative art. When designers and artists explore and refine their ideas, they spend enormous time in the trial and error process [9], because their current tools provide little support for adding customizable constraints to shape their parametric space.

### 1.2 Research Question

The above problems led to my research goal: *To improve existing tools or create new ones to better support parametric design process, especially for*

*parametric space exploration.*

I hypothesized that, by finding a way to manipulate multiple dimensions of a parametric space at the same time and reduce the ranges of parameter values, I can make the exploration process less painful.

To verify or invalidate my assumptions and better integrate my tool into the creative process, I interviewed parametric designers and generative artists about their working process, especially motion designers and visual artists who play with dynamic visual effects. My interviews then led me to design and experiment on the Motion Palette, a creativity support tool that facilitate the motion exploration of dynamic visual effects.

To focus on one research step at a time, I divided the research goal into two research questions, each led a study in this thesis:

1. *What is the creative process like in parametric design and generative art? Especially for dynamic visual effects design, what tools do the designers and artists currently use and how do they explore their parametric space?*
2. *How can I improve existing tools or create new tools to fit in the creative process for dynamic visual effects design? Especially how to support exploration of the parametric space?*

## 1.3 Contributions

My contributions in this thesis come in three aspects: artifact, empirical, and theoretical.

### 1.3.1 Artifact Contributions

Motion Palette is a motion exploration support tool, which is defined by a set of extreme states representing a set of parameter values that control the same visual effect system. By compressing multi-dimensional parameter spaces onto 2D spaces of polygon shapes, it allows exploring multiple parameters at the same time in a visual way. Users can define their own Motion Palettes and save for later use.

### 1.3.2 Empirical Contributions

Parametric designers and generative artists share some common elements or steps in their creative process: they go through a trial and error process due to different types of constraints; they extract and combine, adapt, and

repurpose elements from reference resources or projects; they need to reuse presettings, templates, and filters; they struggle to cross learning gaps due to procedural operation to reach a certain effect; they render in 3D software and polish in 2D software to get static visual results more quickly.

### 1.3.3 Theoretical Contributions

The Motion Palette design concept can be applied not only to motion design, but also extend to other multi-parameter space exploration scenarios. It sets a possible design paradigm for visual directed parametric exploration support tool.

## 1.4 Structure of the Thesis

The rest of this thesis is organized as follows.

Chapter 2 gives the background, mainly looking into literature that strive to solve the aforementioned problems, as well as instrumental interaction design guidelines and examples that inspire my solution.

Chapter 3 introduces the main methods adopted in this thesis: critical object interview, thematic analysis, experimental design and structured observation.

Chapter 4 starts the research with an empirical user study on how parametric designers and generative artists go about their creative process, including their current tools in use. It opens up understanding on possible gaps between ideal creativity support toolsets and state-of-the-art ones.

Chapter 5 is a follow-up study on the design and evaluation of the new motion exploration support tool Motion Palette, based on design implications extracted from the first study.

Chapter 6 concludes the thesis contributions and points out possible ways to extend Motion Palette and the notion of it to other scenarios.

## Chapter 2

# Background

### 2.1 Research Solving the Problems

#### 2.1.1 Disparity Between Tools and Creations

To solve the disparity problem, research has been conducted in running real-time simulation, enhancing node based visual programming tools, and reducing manual adjustments of parameters.

##### 2.1.1.1 Running Simulation in Real Time

In B. Victor's talk, *Inventing on Principle* [30], he proposes to eliminate the gap between writing the code and running the simulation, resulting in tools much like what we see in live coding websites such as CodePen.io [10]. In a game demo that he shows how to control character movement, he suggests to map time space parameters not only for the character real time position, but also its past and future trails. As the ranges of the parameters change, trails of past and future movement also change accordingly in real time.

This approach only makes it easier to locate the parameter that controls the corresponding effect, but it is still painful to manage text based programming.

##### 2.1.1.2 Enhancing Node Based Visual Programming Tools

Image Graphs [18] proposes a novel approach to show intermediate results in node based visual programming tools. By encapsulating code into nodes or patches, users can manipulate modules of functions and parameters, easier than diving into text based code. Then by visualizing each of those nodes, they no longer need to test out each node of operation but directly see the

intermediate result. This design has been widely adopted in 3D rendering tools such as Shader Graph in Unity [27] and Slate Material Editor [2] in 3ds Max.

But the bottleneck of trying out different possibilities is still there: the number of combination possibilities explode as the nodes or patches increase in number; while for humans they can only test parameters one by one to see the impact of the parameter being changed.

### 2.1.1.3 Reducing Manual Manipulation of Parameters

#### Direct Manipulation on the Results

Kazi et al. introduce Kitty [16], a sketch-based tool for authoring dynamic and interactive illustrations. Later, Xing, Kazi et al. turn the tool into Energy-Brushes [33], particularly for illustrating stylized elemental dynamics, e.g., the motion of hair. By drawing lines, the motion type and movement direction of the wind are reified, intuitively specify how the hair would move. To extend the hand drawing manipulation to 3D, Kazi et al. bring up DreamSketch [17], which bridges the ambiguity of early stage sketches and precision of renderable 3D models, changing the way of creating 3D meshes by controlling parameters into direct hand drawing. Jacobs et al. come up with Para [15], a direct-manipulation tool that supports procedural drawing by grouping similar shapes together with constraints that allow live, non-linear, and continuous manipulation with control anchors.

#### Result-Driven Exploration

Design Galleries [20] can automatically generate perceptually different graphics or animations, if an input vector is provided. Users can pick results directly from the galleries, instead of manipulating parameters and wait for generation. Bruckner et al. [9] design a tool very similar, but for dynamic visual effects that change over time, presenting galleries of sample sequences that can be searched and blended. Zsolnai-Fehér et al. teach an AI to synthesize materials [34]: it allows users to assign scores to a gallery of generated materials, then learns the high-scoring samples and recommends many new materials.

These solutions work well for 3D animation rendering that requires heavy computation: they either compute samples in advance, or entirely bypass the computation by simulating a rendered picture with AI. However, the shortcoming is too little direct control in terms of parametric space customization.

### 2.1.2 Little Support for Exploration

Various tools offer support by helping artists progressively add customized constraints to shape the parametric space, as they explore and refine their ideas. For example, Attar et al. [1] innovatively combine physics simulation and 3D modeling together to interactively find the form or shape of the 3D models. Smith et al. [26] suggest that a continuous shape space can be defined by registered models created from examples, and criteria for the space can be set from physical properties and consumer-provided scores, thus customized. In the Gaussian Material Synthesis paper mentioned above [34], Zsolnai-Fehér et al. allow users to score the AI synthesized material, so the AI can learn user preferences and generate materials to their likings.

For live performances, the systems should allow users to freely explore in real time within the range of parameter presets or pre-baked visual effects. Muller et al. [22] design a node-based live changing visual editing system for VJing. Their system values presets-based improvisation during the performance. In the talk *Stop Drawing Dead Fish* [31], Victor divides motion into atom level motion and upper level behavior. He argues that for animation creation, some routine motion can be calculated by algorithms, other motion requiring more nuance can be manipulated spontaneously by hand. With his system, users need to design both levels of motion in advance, but the system leaves space to improvise interactively in live performance.

However, none of these projects ends in a user interface fluid and flexible enough, which can be a crucial demand for exploring continuously transitioning dynamic visual effects.

## 2.2 Research Inspiring My Solution

### 2.2.1 Instrumental Interaction and Substrates

In 2017, Beaudouin-Lafon proposed a new conceptual model towards unified principles of interaction [6] to combine the concept of instrumental interaction and information substrate. This model has inspired the design of my solution.

#### 2.2.1.1 Information Substrates

A substrate [6] is a digital computational medium, usually with a certain structure, that holds digital information. Maudet et al. [21] apply that notion to graphic designers and bring up the concept of graphical substrates, the surprisingly sophisticated structures far beyond grids used by graphic designers. By graphical substrates, they refer to what graphic designers call



“architectures”, “structures”, “rules” or “constraints” that guide the layout, but barely appear in the final form of their work.

In my solution, the Motion Palette is a substrate that defines a structured parametric space and facilitates the exploration of parameter combinations.

### 2.2.1.2 Instrumental Interaction

To interact with substrates, we need instruments that are able to directly operate on the substrates. The interaction instrument [5] is a mediator between the user and the objects of interest. To manipulate objects of interest more intuitively, the design of instruments is better to follow the principles of reification, polymorphism and reuse [7]. Object-Oriented Drawing invented by Xia et al. [32] extends the notion of instrumental interaction, by objectifying graphical attributes into UI elements named as “Attribute Objects”, bypassing the “mediator” and integrating the instruments into objects of interest, which can be directly manipulated by hand through touch gestures.

In my solution, the elements that form and extend the Motion Palette all follow the principles of designing instrumental interaction.

## 2.2.2 Parametric Space Exploration Tools Applying Instrumental Interaction and Substrates

Interestingly, I find some parametric space exploration tools have naturally integrated these models and principles into their design already.

### 2.2.2.1 Customized Color Space Exploration

Playful Palette [25] borrows the practice from oil painting and water color painting and innovates the digital color picker. I’m interested in the way it structures and blends color to accommodate the needs of self-defined color space for digital painters. A palette is defined by color blobs of different colors and sizes, and the relative position of one to another. The color blending involves some complicated maths, but the idea is just interpolation. This structure is easy but allows flexible variations. If we imagine a complete color space as a cube, with H, S, V as the three axes, a palette is a curved surface inside that cube: thus the color space is customized.

### 2.2.2.2 Engineering Parametric Space Exploration

In his digital essay *Up and Down the Ladder of Abstraction* [29], Victor comes up with an approach to interactive visualization that helps the exploration and understanding of an engineering system. What directly influenced

my solution is the interaction of mouse hovering: multiple parameters can be bundled together and the change in the mouse position affects all the parameters bundled. This interaction has naturally applied the notion of substrates.

### **2.2.2.3 Character Facial Space Exploration**

Modern animation software design has integrated the study of how bones of the face, head, and neck affect the movement of the muscles [12]. In facial rigging, animators can move the joints on the bones and the muscles will move accordingly.

In Maya, animators can also define basic facial expressions and blend them with BlendShape Deformers [3]. In practice, some animators actually define basic facial expressions based on emotions. This has inspired me to rethink about motion and its possible upper layers, as well as how they may affect each other.

## Chapter 3

# Methods

For the whole research project, I start with a field study of interviewing parametric designers and generative artists. Then I analyze their behavior patterns during their creative process to find design opportunities to either improve their current tools or invent new tools. After that I brainstorm to several design directions and come up with a final design concept. I then prototype under that design concept and iterate a few rounds to reach a more polished version. In the end, I go back to the users and observe how they would interact with my solution, and find possible improvements to be made.

### 3.1 Critical Object Interview

To better understand the creative process of the participated parametric designers and generative artists, I use a critical object interview [19] technique to encourage them to talk about one or two of their recent projects, recalling step by step how they designed the piece of work, from their initial ideas to final creations, which might involve intermediate artifacts in the process as well.

During the interviews, participants are requested to show me the artifacts and creations, along with the tools involved, whether digital or non-digital. They are the so-called critical objects, making sure that the creative steps recalled are grounded in actual practice. It is even better if they can also reproduce the steps for creating the work. The interviews help me get raw data on what they actually did in the projects, and some thinking behind doing, as well as later reflections. The data collected are more reliable than mere thinking and talking.

The aim of the interviews is to find gaps between what the designers and

artists want to achieve and what they can achieve with their current tools.

## 3.2 Thematic Analysis

For the qualitative analysis of the user studies, I use the thematic analysis method [8] to analyze behavior patterns. For example, for the first interview study, I consider the creative process of each project as a user story. I sort out the creative steps from each story in a temporal order. I then extract codes, key words or short summary, from each creative steps. I conduct cross-reference checks to compare similar creative steps across different user stories. After completing extracting codes from all user stories, I merge similar codes and group codes together to form themes.

## 3.3 Experimental Design

To test out the newly designed UI technique, I design a quasi-experiment to compare it with the traditional UI, i.e., Sliders. I do not hold hypothesis, but rather focus on qualitative analysis of user behavior.

### 3.3.1 Within-Subject / Between-Subject Design

The main aim of the experiment is to test which of the 3 UI conditions (Sliders, Palettes, Sliders + Palettes) is more helpful to support motion exploration, in terms of both productivity and creativity. So I use a within-subject design to test the UI factor.

At the same time, I recruit participants under different expertise, to see the difference in preferences among novice users, intermediate users and advanced users.

### 3.3.2 Quantitative / Qualitative Measures

The experiment is divided into 2 tasks. Task I is a productivity task, to reach a certain visual effect. Task II is a creativity task, to freely explore a preferred visual effect. For the productivity task, I test with variations of the traditional usability testing measurements. But I do not think it is enough to end with quantitative data. It is important to look into user behavior and thoughts to find improvement indications for the newly designed prototype. This is especially true for the creativity task.

### 3.4 Structured Observation

Therefore, along with the quasi-experiment, I conduct a structured observation [14]. I also ask the participants to follow a "think-aloud" protocol: say whatever comes to their mind, including what they are looking at, thinking, doing, and feeling. This helps me to dig deeper into the reasons why they choose certain strategies, or hold certain preferences.

## Chapter 4

# Study One

### 4.1 Interview Study

I am interested in the strategies, tools, and techniques used by parametric designers and generative artists, when they create digital artifacts that are controlled by parameters and thus not directly manipulable.

#### 4.1.1 Participants

I recruited 10 designers and artists (8 male, 2 female), with 1-5 years of experience. They use different software to build digital artifacts, ranging from 3D animation, 2D and 3D special effects, and interactive new media installation, to architecture, industrial design, and 3D printing. I also gathered data from 1 documentary of an architect.

#### 4.1.2 Procedure

I conducted 10 semi-structured critical object interviews with the participants in their office or in a quiet environment for about an hour. Each participant talks about their recent projects, including how they turn their initial idea to its final form, problems they have encountered in the process, and workarounds to appropriate existing tools for their customized needs.

#### 4.1.3 Data Collection

I took audio and video recordings of the interviews, including screen recordings of the participants showing the digital artifacts and operating the software on their computers. I took notes of key points and photographed the artifacts and the illustrations from their explanation in the interviews.

#### 4.1.4 Analysis

I conducted a Thematic Analysis to all the data collected, including extracting keywords or codes from each interview, and grouping similar ones together and finding a common theme for each group.

### 4.2 Results and Discussion

#### 4.2.1 Thematic Analysis

From the Thematic Analysis I found the following categories of common elements or steps in the participants' creative process.

##### 4.2.1.1 Constraints: Trial and Error

We found for more engineering oriented design projects, which aim to be produced in materialized form, such as industrial design and architecture design, the entire projects are more function directed, including a lot of functional and usage constraints, as well as material and production constraints. For example, P11 wanted to simplify a concept model of a 39-story tower building to make it constructible while keeping its visual form. He faced the challenge of making the model controlled by as few parameters as possible, while meeting the needs of constantly changing indices, coordinating the shape of a specific layer and all functional areas on that layer, eliminating bending inclined columns, etc. He needed instant visual feedback of all these constraints to do better fine tuning for the final form of the model.

While for more artistic projects where designers and artists have more freedom on deciding the final forms, the whole projects are more visual directed, and the designers and artists tend to have self-imposed constraints, mostly in aesthetic aspects. For example, for P7, he wanted to explode 3D models along the Z axis and 3D print them in rectangular prisms. His project was more of a research and design project with more freedom to explore whatever forms that have the most satisfying visual outcomes. To keep the series in a coherent form, he decided himself to print the explosions in the same transparent rectangular prisms around 6" tall, 3/4" wide in each direction, which lead him to adjust the different 3D models many times to fit in this cuboid. (See Figure 4.1)

With the above constraints, the process of trial and error is inevitable. For a single parameter, for example, in P7's case the position of model fragments along the Z axis, at the very beginning was only a vague idea to test if it would generate interesting results. After testing with different clustering

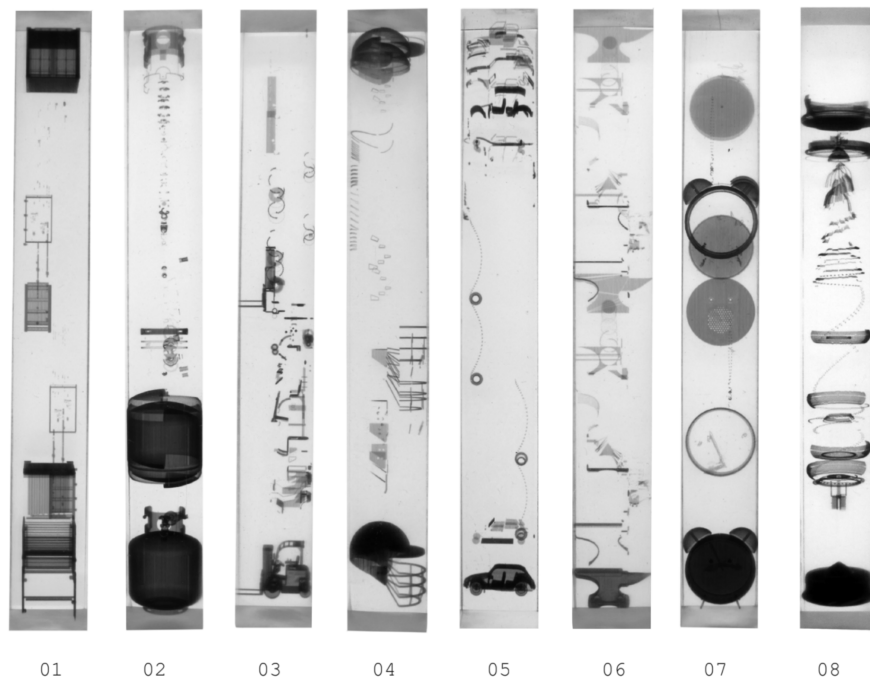


Figure 4.1: Explode 3D models along the Z axis

of the same model and testing among several different models, he found some results were boring and ruled out, while others were revealing many more insights on the structure of the models, leading to fascinating visual outcomes.

For the example of combining several parameters, P7 also created an installation of a CRT computer screen and a camera on a rotating plate to make electronic drawings. He also developed an electronic circuit to control the drawing process. For exploring the combination of parameters, he slightly adjusted 1 parameter at a time while keeping all other parameters intact, and listed all variant results to compare and choose the final form. It is a strategy to explore as many possibilities as possible. (See Figure 4.2)

#### 4.2.1.2 References: Extract and Combine, Adapt and Repurpose

Almost all projects involve references. It is a way to save the energy of reinventing the wheel, as well as a flexible tool to spark new ideas. For the former, starting from scratch takes up too much time on coming up with



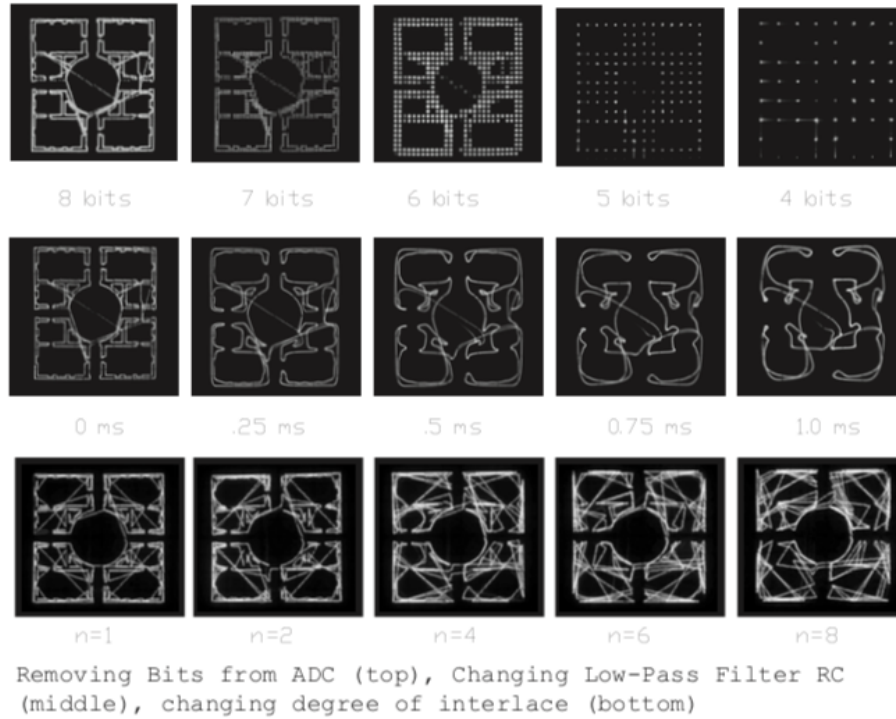


Figure 4.2: Test parameters one by one to list all variations

new solutions to existing problems. For the latter, sometimes repurposing an existing solution to another similar or related problem will generate unexpectedly clever design.

From my interviews, projects can be function directed, visual directed, ambience directed and motion (interaction) directed, 1 type or many at the same time. For function directed and visual directed aspects, the references resemble much more to their final forms. For example, P6 designed a lamp that were inspired by the crane on construction site, so the form of a tilting arm is taken as a reference for the lamp arm. Another example would be P1, who searched for a corner of a village in Google Maps and referred to it for 3D modeling in his animation.

For ambience directed and motion (interaction) directed aspects, however, a lot of adaptation is needed to fit the references in with their new projects. P2 needed to communicate with the client and let them pick their wanted vibe for the opening title video of a TV show, so he provided several simplified moodboards to the client. After the moodboard was chosen, he had to come up with a completely new script for the animated scene and synthesize key

elements out from the script, and the final visual effects looked similar to the chosen moodboard only in the sense that the ambience also feels warm like home. P3 was initially inspired by a project designed for mouse interaction, in which the mouse can control an invisible ball to interact with some vertical strings. He wanted to make an interactive installation with Kinect, but keep the form of the ball interfering the movement of the strings, so he tried to combine another project with body gesture control by replacing the control of mouse with body gesture. To keep coherent with the theme of the project, he had to redesign the body gesture as well as refine the effect of the interaction, in his case fixing the two ends of each string and making the strings more elastic.

Repurposing is used for not so directly related references. For P5, her project was to design a Culture and Art Center building at a road intersection. One of her teammates was into paper cutting, so they took the stretched shape of a rectangular paper interlaced with 7 cuts (Figure 4.3), and transformed it into the building's final topview form.

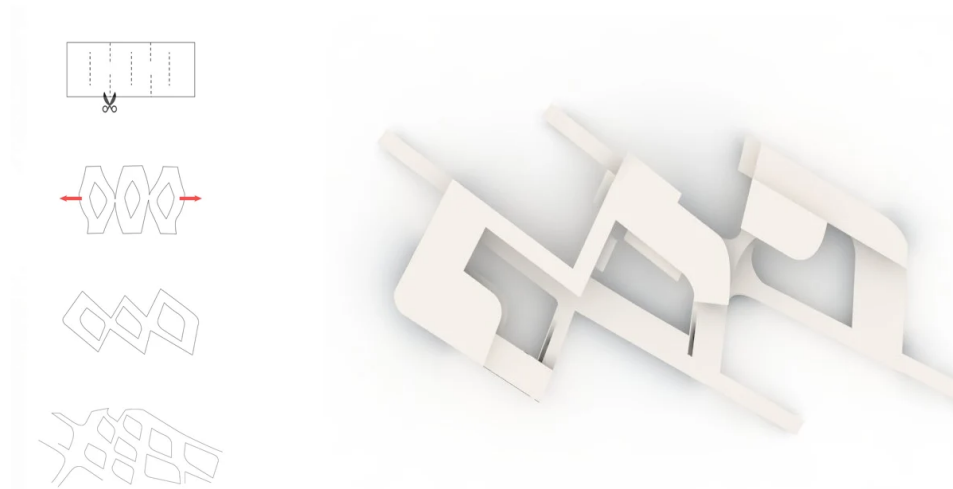


Figure 4.3: From paper cutting to architecture design

#### 4.2.1.3 Presetting, Template and Filter: Reuse

In most visual programming software, only atom level tools are provided, designers and artists need to build their own customized upper level toolbox.

For a simple example, P4 as a 3D modeler would have projects that contain similar materials. In the software 3ds Max she used, the presetting of brass is defined, so when she wanted a gold material, she could adjust the parameters of brass to tune into gold. Once her gold material is saved, she can reuse it as many times as she wants. The same applies for global illumination settings, which contain more complex parameter combinations.



Figure 4.4: Watch advertisements created by the same 3D modeler

For very similar projects that contain a lot of similar elements, a template would be very helpful. P4 made 3D models for different watch advertisements. She used very similar materials, such as glass, silver and leather (See Figure 4.4). Every time she remade those materials. If she can define a watch template that contains all these materials, the process of rendering similar watch models would require much less redundant work.

To dig deeper into the parameters, there can be filters. Current 3D software do not seem to have this concept, but in 2D image processing, this is not a new invention. In Photoshop, the hue, saturation and brightness of one image can be matched to another. Or better, take the example of the everyday app Instagram, there are predefined image filters: with a simple tap, the selected filter is applied to the image being processed. Basically they analyze the graphs of certain parameters, and tune the value within a certain range.

#### 4.2.1.4 Procedural Operations: Struggle to Cross Learning Gap

The procedural operations of achieving a certain effect are hard to be found out and internalized by beginners. They are not intuitive at all and cause the learning gap that prevent users to gain a deep understanding of the system: without following the steps in tutorials, beginners would hardly find out by themselves those steps. This is confirmed in our interviews: beginners of certain effects usually learn tutorials first, and then extract the

elements they need and adapt to their own projects. This also applies to P1, an advanced special effects artist. When he wanted to make a new effect that does not resemble any of his previous work, he chose to learn a tutorial rather than try out by himself. The reason behind is that the procedural operations scattered in parameter manipulation are hard for him to gain a coherent understanding of what the software is capable of, only via testing out parameter combinations on his own. Parameter manipulation in most software are designed as controlled by sliders or text fields in blocks of properties, whether in the properties panels, or in the visual programming patches or nodes.

Reification of the properties (if possible combine some to build integrated tools), would be much more intuitive compared to text labels. Some properties will fall into the categories of instruments, others will be substrates. For example, P11 wanted to make the 39-story tower building constructable, and he finally simplified the shape of each layer in the conceptual model to be only controlled by 4 corner radius. The corner radius of the neighboring layers need to transit smoothly. With the help of Bezier tool, which is an instrument that allows direct manipulation, the smooth transition is easily guaranteed. For the representation of different function areas inside each layer, which are substrates to ensure the area for different functionalities, an instant visual feedback is provided to suggest if the shape of the entire layer can hold every function area within.

#### **4.2.1.5 Tool Stack: Render in 3D and Polish in 2D**

If the final output is in 2D form, either still image, or 2D video, 3D designers and artists will balance the amount of time in 3D software and 2D software. Usually they tend to put the polishing of the visual effects in 2D software, as it is much quicker without having to wait for simulation and rendering.

### **4.2.2 Design Implications**

From the analysis I found the following design implications to integrate some of the design practices into the software tools.

#### **4.2.2.1 Constraints: Trial and Error**

For visual programming tools, the bottleneck for combinations of patches or nodes is that as the number of nodes increases, the possibility of combination grows at an exponential rate, known as “factorial explosion”. Without instant visual feedback for each parameter, designers and artists need to test

parameters one by one and run the simulation to see the combined result. Intermediate level of representations would be very helpful to provide such instant feedback.

For designers and artists who focus on parametric design and generative art, they tend to have this pattern in their creation: they have a rough idea at first, and then explore around it to see possibilities all around it nearby, rule out unwanted ones and keep some interesting ones, by trial and error they finally lock up on the final decision. For some designers and artists who do series of work, they need this exploration to make multiple interesting combinations to form a series. The automation of this exploration process would help them find new combinations that are otherwise easy to miss out in their manual design space exploration.

#### **4.2.2.2 References: Extract and Combine, Adapt and Repurpose**

The frequently adopted practice of making references suggests an easier way to extract and combine different elements from existing projects would be helpful.

Adapting and repurposing as design strategies can be applied in the design of our new tool. We can borrow concepts from other design software to our target design tool, for instance, borrow the concept of filter from 2D and apply to 3D, or find proper metaphors to reify instruments and substrates, for instance, visualize force field as vectors that can be directly manipulated.

#### **4.2.2.3 Presetting, Template and Filter: Reuse**

Presetting, template and filter can be considered 3 different levels of parameter combinations. Presetting is on the level of 1 or several nodes of parameters to reach an atom effect; template is on the level of a whole package to complete a certain scenario; filter is a combined value constraint for 1 or several parameters together to keep the value within a certain range and rule out the rest.

Reuse of the existing parameter values should be able to be applied on the presetting level, template level and filter level. Different levels of reuse will make the iteration of the creative process easier. An intuitive way to keep the snapshots of history would also be helpful to go back to a trial and start from there again.

#### **4.2.2.4 Procedural Operations: Struggle to Cross Learning Gap**

Making the procedural operations more visible and intuitive by reifying the parameters according to their meaning and usage. As mentioned in repur-

posing, we can also try to find proper metaphors to adapt indirectly related tools to our intended new design.

## Chapter 5

# Study Two

### 5.1 Design Process

#### 5.1.1 Brainstorming

According to the design implications concluded from Study One, I came up with 3 ideation directions.

##### 5.1.1.1 Direct Control

State-of-the-art node based visual programming tools have intermediate visualization of each node, but few of them have enabled reified controllers that allow direct manipulation on the visual results. The manipulation can even be backwards: right now only the parameters control the generated visualisation, but there can be manipulations on the visualisation that affect backwards on the parameters.

##### 5.1.1.2 Stylistic Toolbox

I learned from interviews that the procedure steps to manipulate a series of parameters are not intuitive enough to be discovered and internalised. Even for designers with some years of experience, they still need to learn tutorials step by step to achieve a certain effect. There can be templates of workflow and pre-settings to fill in the learning gap for beginners. For expert users, they can also define their own templates and filters. Filters are an upper level metrics of parameters that help tune the value within a certain range.

### 5.1.1.3 Customized Exploration

Playful Palette [25] tries to help painters define a customised color space, either in the form of blending color blobs, or in grids of colors, to explore easily and exploit in paintings of their own style. So the same idea can be applied to motion, and I can help designers to define a customized motion space, from atom level parameters, to upper level behaviors.

## 5.1.2 Converging Ideas

I am personally interested in the ideas of customized exploration, especially how to help visual artists explore motion for dynamic visual effects. Besides, researchers addressing parametric space exploration problems mainly focus on more static art or design forms such as 3D modeling, which leaves opportunity for designing tools to explore dynamic motion. In Study One, I have interviewed plenty of motion designers and become familiar with their strategies in exploration and their consequent workflow. So I decided to further develop ideas in this direction.

### 5.1.3 Use Scenario

To prototype for the possible system, I extract user traits and task workflows from interviews in Study One and create a persona artist who would benefit from my solution.

Edward is an visual artist who creates dynamic visual effects. His daily tools for creation include Unity and WebGL. His work is characterized by irregular movement of particles and other dynamic visuals. To create his artwork, he deals with sliders in properties panels and text based programming on a daily basis. To explore different possibilities of one digital art piece, he has to save a lot of screenshots to record all the parameter values and the corresponding effect. He also works as a VJ to do live performance at music events. For the live performance, he would use a MIDI controller with sliders that are mapped to the parameters in his dynamic visual artwork.

### 5.1.4 Design Concept

I started designing a concept with sketches on paper (See Figure 5.1) to list whatever features that might come useful in the prototype, including blending different types of motion together, categorizing motion with different ranges of parameter values combined, heat map for the palette to indicate



user’s interest in different types of motion, taking snapshots of history for future revisit, etc.

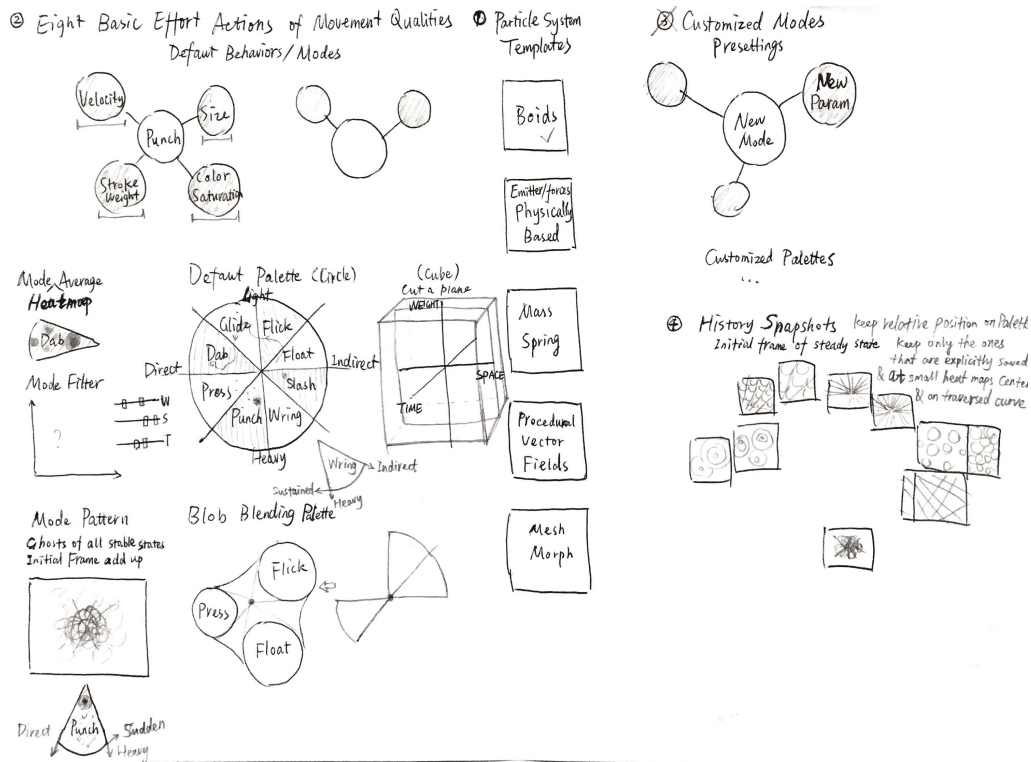


Figure 5.1: Sketches on paper

Then I tried to develop a coherent design concept (Figure 5.2). A palette can be defined by several extreme states. I can place them in whatever order or distance I like. All the discrete extreme states can form a continuous space, visually represented as a polygon. Inside the polygon space, a point represents an interpolation state. The parameter values of the interpolation state can be influenced by all the extreme states, depending on how far away the point is to the extremes: the closer it is, the more it is influenced.

To further that concept and see what elements will show on the user interface, I tried to think in the shoes of the persona Edward, mentioned in Use Scenario, and drew a sketch with an online prototyping tool Figma [13] (Figure 5.3). The final design concept contains the following elements: to support exploration, a heat map is shown on top of the palette; to revisit exploration history, snapshots of states can be saved; to reproduce a series of states, a path of exploration can be recorded.

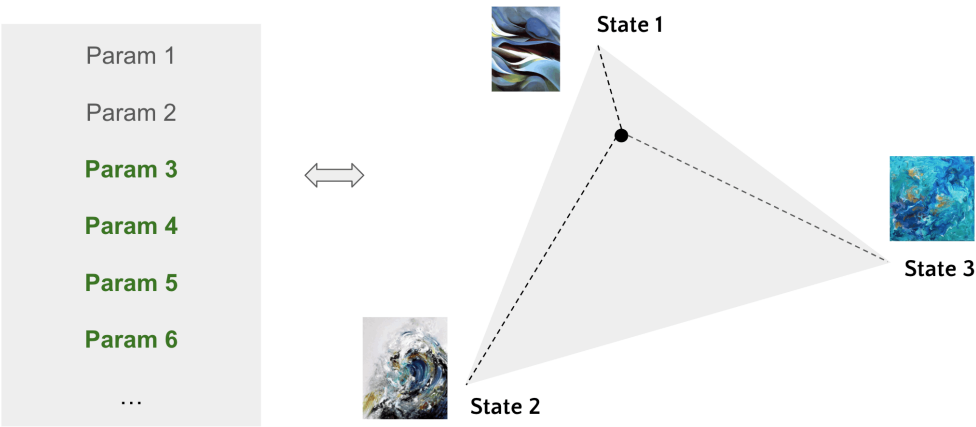


Figure 5.2: Design concept

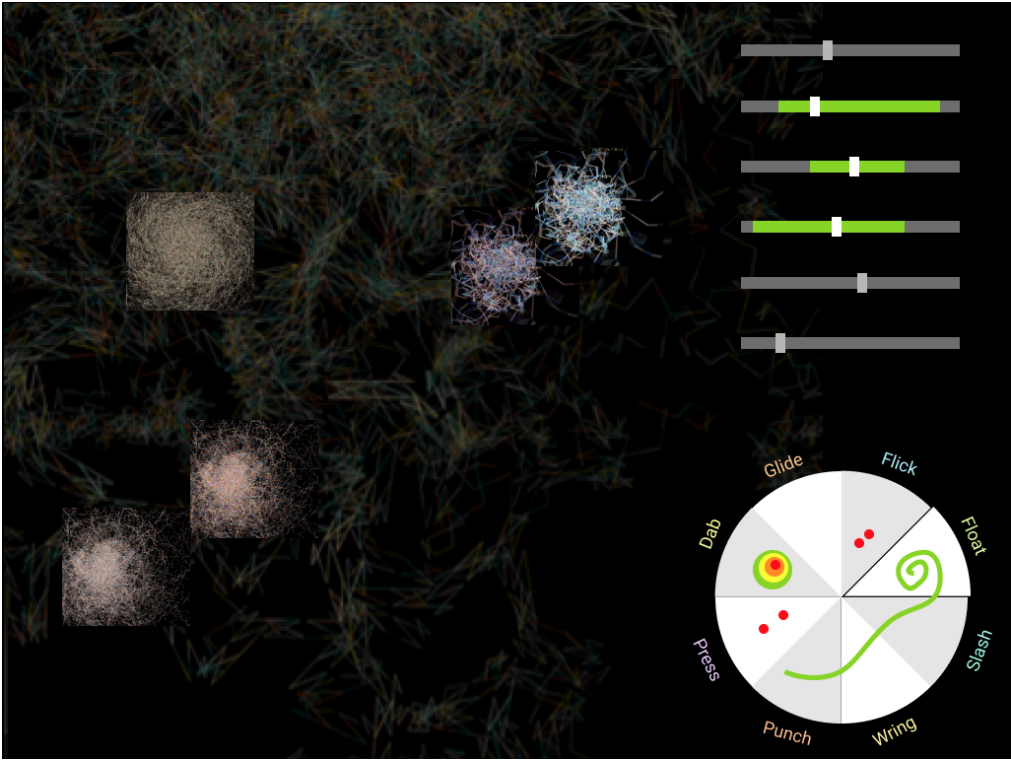


Figure 5.3: Design concept with UI elements

### 5.1.5 Prototyping

I built a paper prototype (See Figure 5.4) to try out possibilities of interaction with the objects of interest, represented by each small piece of paper or a drawn element on the paper user interface.

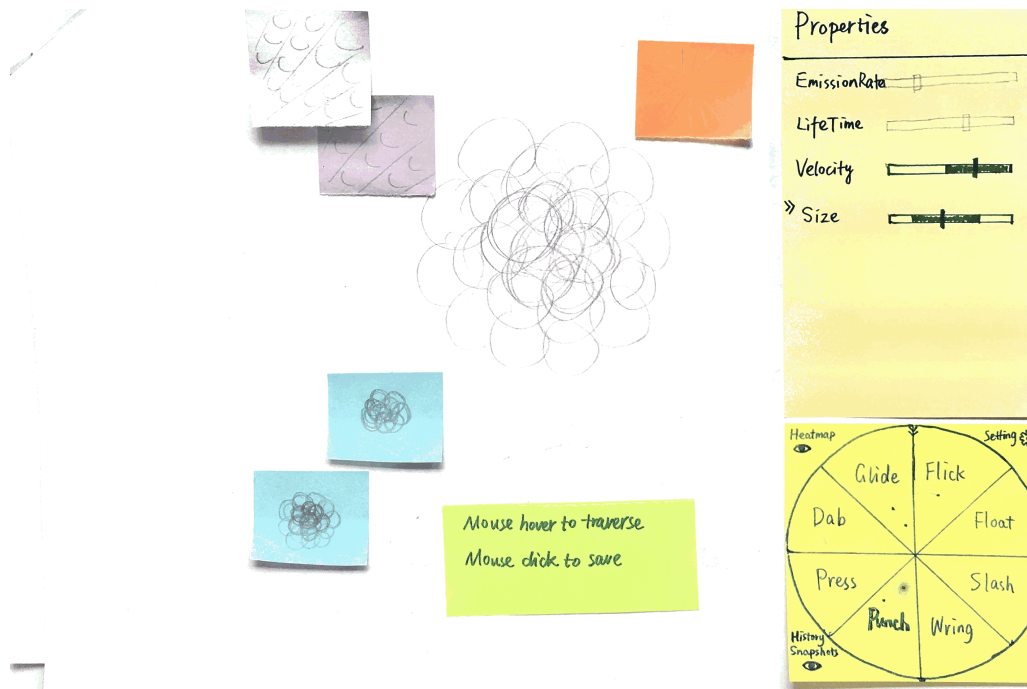


Figure 5.4: Paper prototype

### 5.1.6 Implementation

After that I tried to implement the very basic element of interaction in the design: the palette that can control all parameters at the same time (See Figure 5.5). I started to build the digital prototype with an openFrameworks example of a particle system, from Perevalov's book *Mastering openFrameworks: Creative Coding Demystified* [24]. I kept the parameters and the visual effect, and added the palette feature containing dots of extreme states (red dots in Figure 5.5). When the mouse cursor hovers inside the palette (red circle in Figure 5.5), the parameters are recalculated according to how far away the mouse cursor is to each of the extreme states.

The recalculation causes the change in the current state of the visual effect, which means the position of the mouse cursor controls that change.

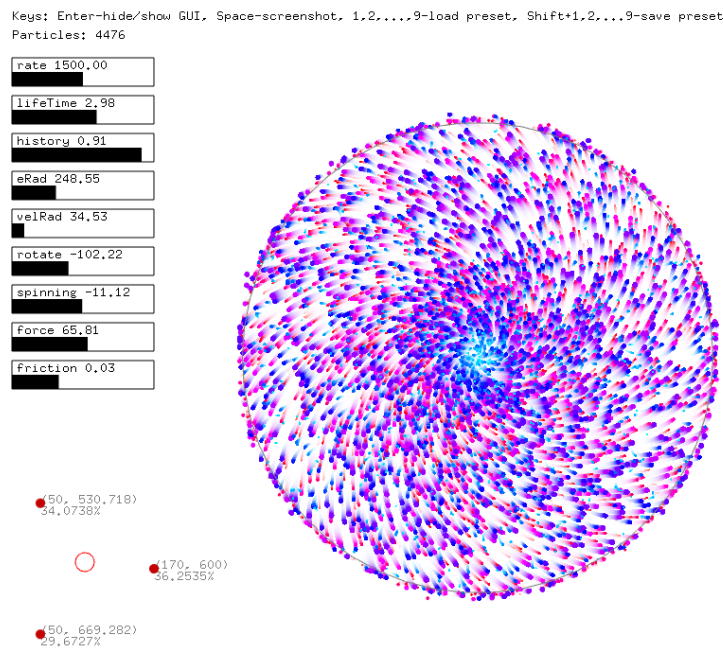


Figure 5.5: Simplest digital prototype

In my design, I would like the farther away the mouse cursor is to an extreme state, the less influence the extreme state has on the current state of the visual effect.

The current state is in fact an interpolation state of all the extreme states, that is to say the current parameter values are interpolations of parameter values of the extreme states. This led me to find an algorithm called Inverse Distance Weighting [4], which calculates the interpolation in the way that suits my design.

After another few rounds of iterations, the final digital prototype moves the palette to the center of the canvas, and adds many new features (See Figure 5.6).

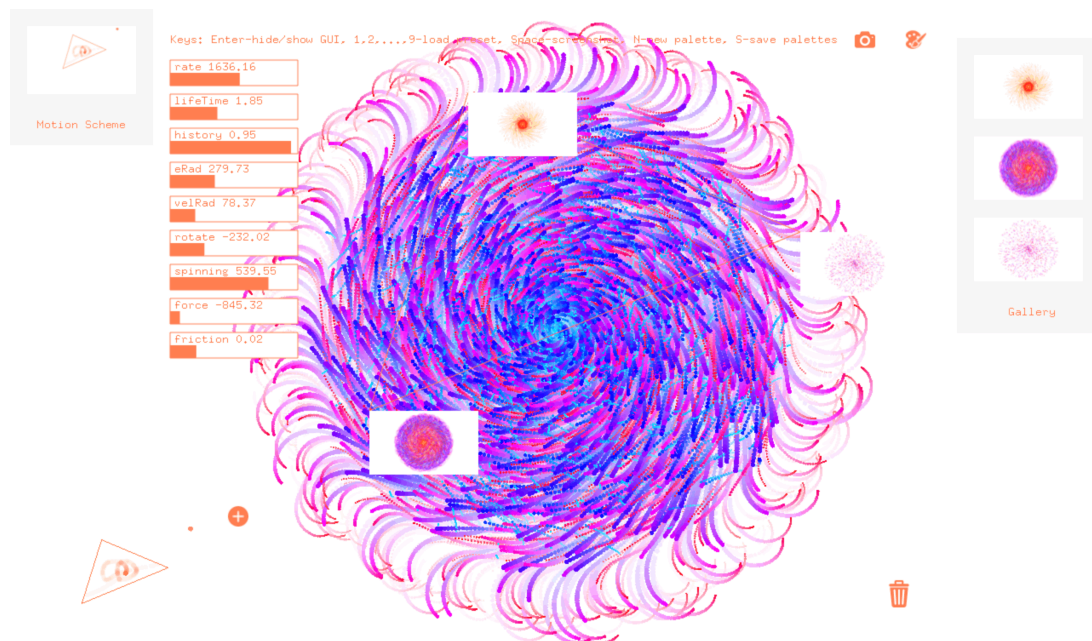


Figure 5.6: Final digital prototype

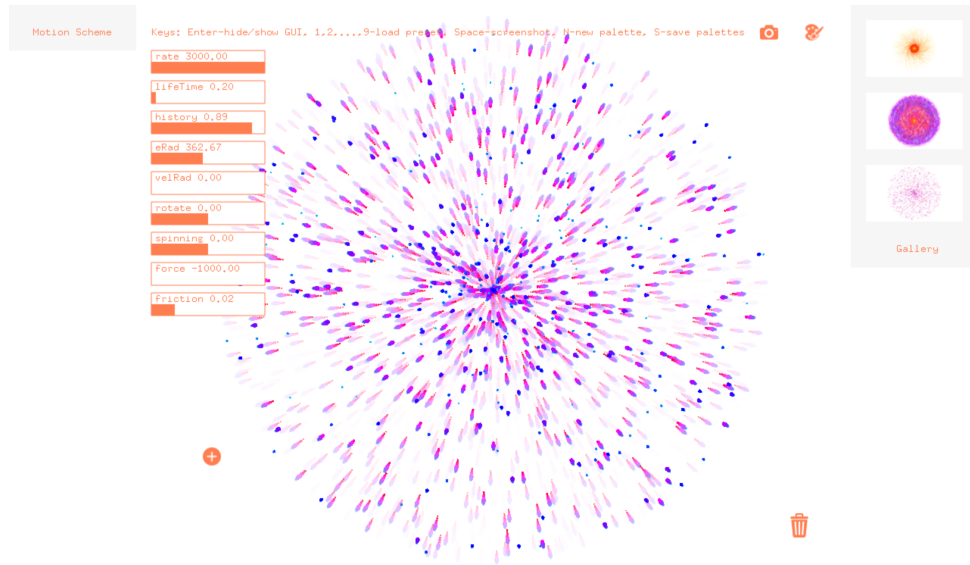


Figure 5.7: Saved states in gallery

## 5.1.7 Features

### 5.1.7.1 A Gallery of Saved States

Users can manipulate parameters with the Sliders. The visual effect will change in real time according to the values of all the parameters. At each moment, the system state is determined by all the parameter values.

When the users are satisfied with the current visual effect, they can save it along with the all the parameter values in the Gallery. A thumbnail will appear in the Gallery, representing the saved STATE (Figure 5.7). The users can revisit the saved state by clicking on it. Once a STATE is clicked, the system restores to that state, namely, all the parameter values will change to the saved values, and the visual effect will change accordingly.

### 5.1.7.2 State Copies that Form Palettes

The users can drag the states out from the Gallery. Once a state is dragged, its copy is created on the canvas, where the visual effect is drawn (Figure 5.8). The users can select the STATE COPIES by clicking on them, and generate a Palette by clicking on the palette button. Once the STATE COPIES form a Palette, they become part of the Palette as EXTREME STATES. Multiple palettes can exist at the same time on the canvas.



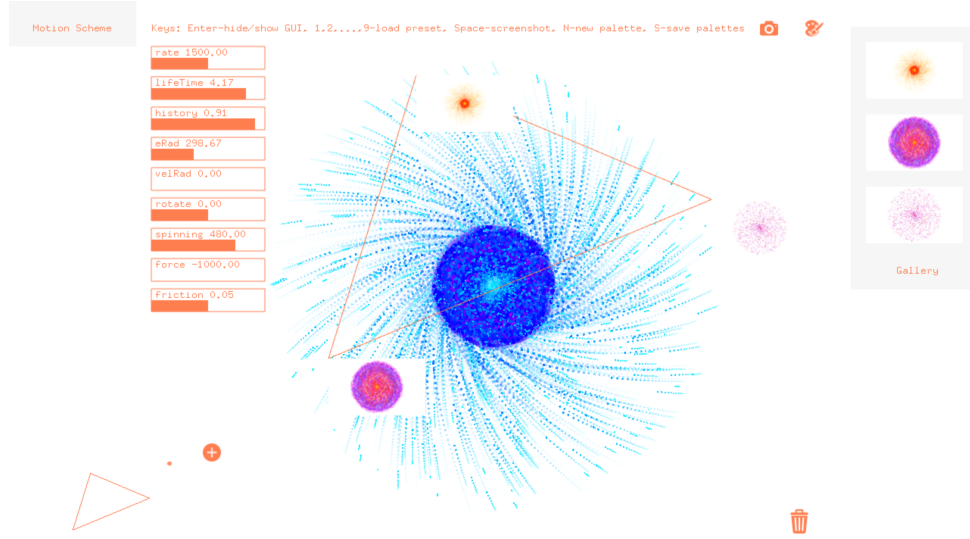


Figure 5.8: State copies that form palettes

The users can hover inside the Palettes with the mouse cursor, and the current interpolation state will be recalculated according to the position of the cursor relative to the position of all the EXTREME STATES that form the Palette.

If the users want more interpolation states (higher granularity) inside the Palettes, they can drag the EXTREME STATES further away from one another.

### 5.1.7.3 Palette Representation

On the bottom left of the canvas, a smaller view of all the Palettes on the canvas is presented as the Palette Representation (Figure 5.9). As the users hover inside the Palettes, the mouse cursor is shown as a red dot in the Palette Representation, and the trajectory of its movement will appear as a semi-transparent stroke in it as well. In this way a heat map of the visiting history inside the Palettes is generated, so that the users can see which areas of the Palettes are more frequently visited, and which areas of the Palettes are less explored.

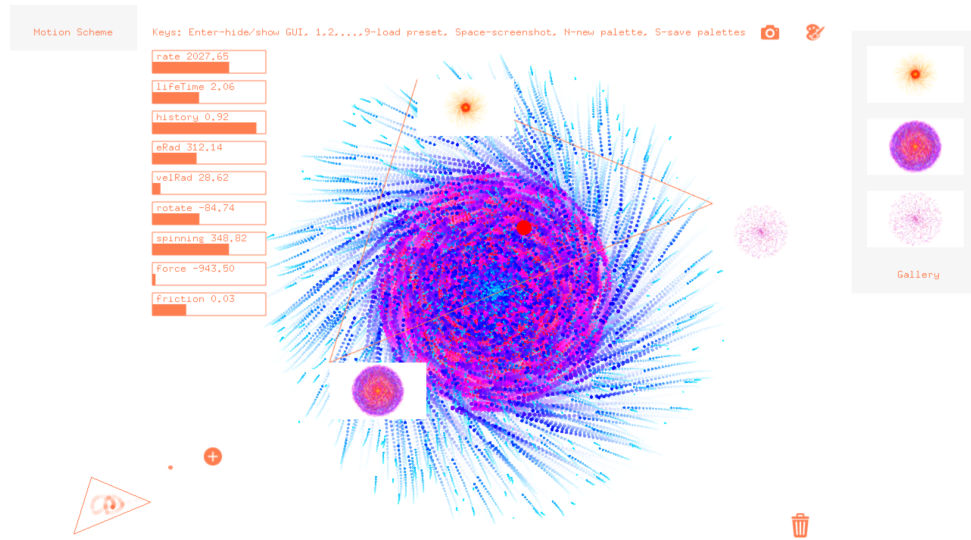


Figure 5.9: Palette representation

#### 5.1.7.4 Palette Snapshots in Motion Scheme Panel

If the users are not satisfied with the Palettes they have created, they can clear all current Palettes by clicking on the trashcan button. On the contrary, if they are satisfied and may want to use the current Palettes in future, they can save them as a PALETTE SNAPSHOT by clicking the plus button on the top right corner of the Palette Representation. A thumbnail will appear in the Motion Scheme Panel (Figure 5.10). Same as the STATE, the users can click on the thumbnail to restore the saved PALETTE SNAPSHOT.



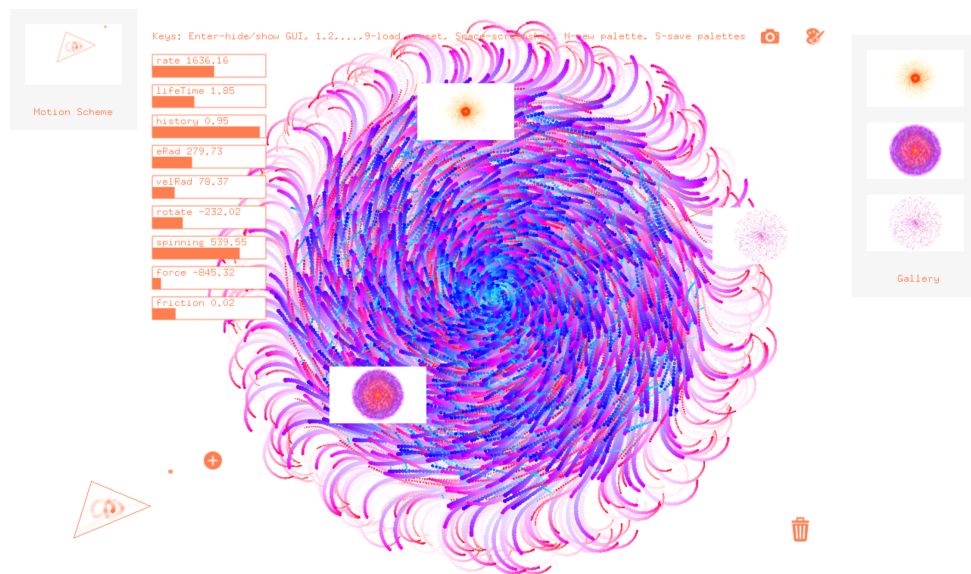


Figure 5.10: Palette snapshot in motion scheme panel

## 5.2 Quasi-Experiment: Structured Observation

A particle system controlled by 9 parameters was implemented to generate visual effects for the experiment. I am particularly interested in the strategies and preferences that the participants take to complete the tasks of reaching a target effect and free exploration. I want to know if the current features of Motion Palette satisfy the needs for basic motion exploration.

### 5.2.1 Experimental Design

I use a within-subject design for the factor UI with 3 conditions/modalities. I also use a between-subject design for the factor expertise.

For both Task I and Task II, in order to avoid order effects, I use a counterbalancing strategy, Latin Square, provided by the experimental design tool Touchstone2 [11]. Sliders, Palettes, Sliders + Palettes are counterbalanced by UI factor across participants. This ensures that two orders between each possible pair of the 3 UI conditions are presented to the participants.

#### 5.2.1.1 Task I: Reaching a Target Effect

Ask each participant to try and reach a target effect (show the participants a video of the target effect).

Considering equivalence, for each UI condition/modality, each participant will be shown a different target effect that's about the same level of difficulty to reach from the same original setting. Target A, B and C should all modify 3 sets of the same number of parameters from an original setting. The extreme states should also be decided by moving the same 3 parameters as the targets from the original setting (keep the other 6 intact).

Participants will draw lots to decide the 3 target effects for each UI condition: Target A/B/C (See Figure 5.11).

#### Factors

1. UI with 3 conditions/modalities
  - (a) Sliders: 1 original setting - a midpoint state (out of 3 extreme states)
  - (b) Palettes: 1 palette (out of the same 3 extreme states)
  - (c) Both the Sliders and the Palettes

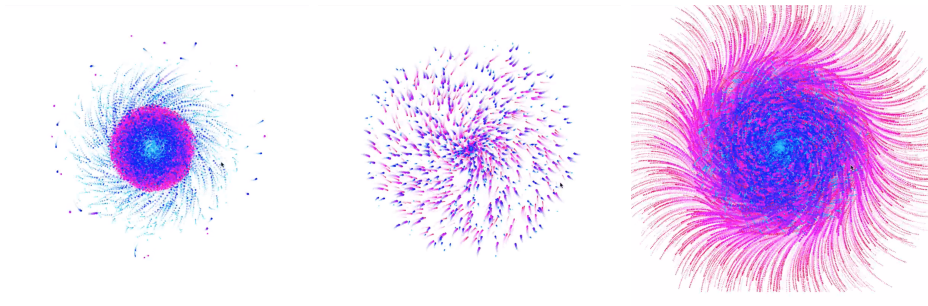


Figure 5.11: Target A, B and C

## 2. Expertise with 3 conditions/modalities

- (a) Novice
- (b) Intermediate
- (c) Advanced

## Measures

### 1. Quantitative measures

- (a) Number of resets and trials
- (b) Time to complete the task
- (c) Distance to the target effect (calculate delta values of all parameters between their final saved state and the target effect)
- (d) Reflection (likert scale)
- (e) Overall ranking (likert scale)

### 2. Qualitative measures

- (a) Structured observation and thematic analysis
- (b) Reflection (short interview)
- (c) Overall ranking (short interview)

### 5.2.1.2 Task II: Free Exploration

Ask each participant to try and create an effect they really like (provide the participants with 9 presets for all 3 UI conditions).

**Factors**

1. UI with 3 conditions/modalities
  - (a) Sliders
  - (b) Palettes: the participants will create their own palettes
  - (c) Both the Sliders and the Palettes
2. Expertise with 3 conditions/modalities
  - (a) Novice
  - (b) Intermediate
  - (c) Advanced

**Measures**

1. Quantitative measures
  - (a) Time to complete the task
  - (b) Reflection (likert scale)
  - (c) Overall ranking (likert scale)
2. Qualitative measures
  - (a) Structured observation and thematic analysis
  - (b) Reflection (short interview)
  - (c) Overall ranking (short interview)

**5.2.2 Participants**

I recruited six participants (2 male, 4 female): 2 with an HCI background (P1 and P2, treat as beginner/intermediate users), 2 with a science background (P3 and P4, treat as novice users), and 2 with parametric design/generative art experiences (P5 and P6, treat as intermediate/advanced users).

**5.2.3 Apparatus**

The experiment was conducted on a MacBook Pro (Retina, 13-inch, Late 2013) laptop. The Motion Palette system is an application that runs locally on the laptop, which is developed with openFrameworks, an open source C++ toolkit for creative coding [23]. The target effects were shown on an extended screen. A mouse connected to the laptop, the laptop trackpad and keyboard were used as the input device.

### 5.2.4 Procedure

Before the tasks, there is a tutorial session. Each participant is shown how the Sliders and Palettes work. Then they are given about 5 minutes to try the user interfaces out and are encouraged to ask questions regarding the usage if they have any doubt. This session is not counted as part of the tasks.

Then each participant is given 2 big tasks aiming at different goals (Task I: reaching a target effect, Task II: free exploration), each with 3 UI conditions (Sliders, Palettes, Sliders + Palettes). So that is 6 small tasks in total for each participant. When they are trying to complete the tasks, they are requested to follow the “think-aloud” protocol: say whatever comes to their mind, including what they are looking at, thinking, doing, and feeling.

After the participants complete each small task, they are asked to fill in a 5-point likert scale post-questionnaire, containing 6 statements about how they feel when they conduct the small task with the given UI condition. Each participant is also shortly interviewed about how they settle their choices.

After the participants complete each big task (consisting of 3 small tasks with the 3 UI conditions), they are asked to do an overall ranking to compare the 3 UI conditions (Sliders, Palettes, Sliders + Palettes) for achieving the same goal, including a 5-point likert scale for each of the UI conditions to state their preferences, and a short interview about why they would think that way.

At the end of the experiment, they are asked to fill in a personal information post-questionnaire, to state their background and expertise, as well as how much experience they have in parametric design or generative art.

### 5.2.5 Data Collection

For quantitative analysis of the objective results, I collected task completion time for both big tasks. I also collected number of resets and trials for Task I, as well as the parameter values of the final saved state to calculate the distance to the target effect in Task I.

For qualitative and quantitative analysis of the subjective results, I collected the following types of data: observation notes, screen and audio recordings of the experiment, answers to questionnaires after the tasks, and audio recordings of the short interviews.

### 5.2.6 Analysis

For the objective results, I mainly listed task completion time to compare the efficiency, sum of resets and trials to compare error rate, and calculated the distance to the target to compare the accuracy of the 3 UI conditions.

For the subjective quantitative results, I calculated the average grading of the 5-point likert scale statements, to compare which of the 3 UI conditions performs best in the 6 assessed aspects to the users. I also counted how much the participants like the 3 UI conditions, to have a general idea on which UI condition was the favorite among the participants.

For the subjective qualitative results, I used the same method in Study One, Thematic Analysis, to extract key words as codes, and combine similar codes together to form a group. Then I looked for a common theme for all the codes in the same group.

Since this is a preliminary study and relies more on the structured observation to get more qualitative results and deeper understanding of how the participants interact with the Sliders and the Palettes, I do not hypothesize on the efficiency, accuracy or satisfaction of the 3 UI conditions.

## 5.3 Objective Results

I collected a total of 36 experimental trials ( $6 \text{ PARTICIPANT} \times 2 \text{ TASK} \times 3 \text{ UI CONDITION} \times 1 \text{ TRIAL}$ ).

### 5.3.1 Task I: Reaching a Target Effect

#### 5.3.1.1 Completion Time (only for reference)

In the experiment, I told the participants that the requirements for the task is to try their best and reach the target effect as close as possible. Task completion time is only for reference.

Table 5.1 and Figure 5.12 shows the completion time (in seconds) for each participant.

UI Condition	P1	P2	P3	P4	P5	P6
Sliders	427	414	519	467	361	264
Palettes	92	226	182	338	286	129
Both	129	445	398	215	258	286

Table 5.1: Task I Completion Time (seconds)

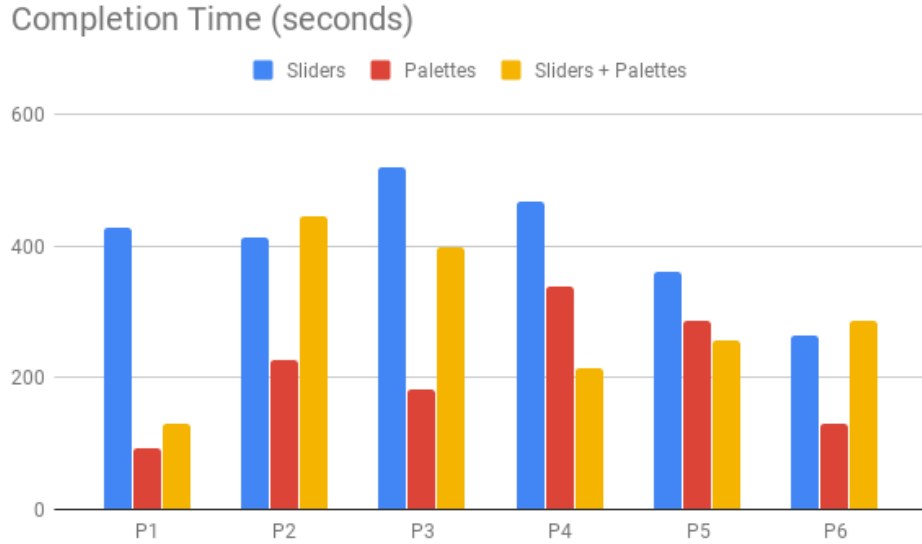


Figure 5.12: Task I Completion Time (seconds)

Least completion time does not necessarily mean the UI condition works most efficiently, as different users have different backgrounds and preferences, and learn new things at different paces. But we can see from Table 5.1 that for the majority of the participants (4/6), it took them less time to reach the target effect if the UI condition contains Palettes.

### 5.3.1.2 Number of Resets and Trials

The majority of the participants (4/6) did not go back to the original setting, even if I told them that the target effect is only 3 parameters away from it. Nearly all of them preferred defining their new anchor states, and adjusted from there. So to have a general idea of errors occurred, I considered both the number of resets to the original setting, and the number of trials from their self-defined anchor states.

Table 5.2 shows the sum of resets number and trials number for each participant.

### 5.3.1.3 Distance to the Target

I recorded the value of all parameters for their final saved state, so that I can calculate the distance of the final saved state to the target effect state. Each

UI Condition	P1	P2	P3	P4	P5	P6
Sliders	2	2	1	1	1	3
Palettes	0	0	0	0	0	0
Both	0	0	2	1	1	0

Table 5.2: Task I Number of Resets and Trials

parameter value is normalized to map the minimum value and maximum value to  $[0, 1]$  for the calculation.

Table 5.3 and Figure 5.13 shows the distance to the target for each participant.

UI Condition	P1	P2	P3	P4	P5	P6
Sliders	1.70	1.87	0.25	0.38	0.74	0.24
Palettes	0.35	0.41	0.47	0.21	0.33	0.52
Both	0.45	1.30	0.69	0.40	0.46	1.40

Table 5.3: Task I Distance to the Target

We can see from Table 5.1 and Table 5.3 that for participants with more rigorous scientific mindsets (P3 and P4), they spent more time to understand how each parameter in the Sliders affects the visual effect, thus their final results manipulated with Sliders are more accurate than other people. P6 is an exception because he has very similar motion design experience of the dynamic particles. He spent the least time to figure out and remember what each parameter controls, and got the closest result.

Otherwise the Sliders do not necessarily help to create more accurate results, which is both a bit surprising and understandable. Although Sliders might seem to be able to control more accurately, not all people are patient enough to memorize the meaning of 9 parameters at the same time, or try them out one by one until they find the right parameter values.

On the other hand, Palettes seem to have less accurate control, but the UI conditions comprising the Palettes yield better results in general.



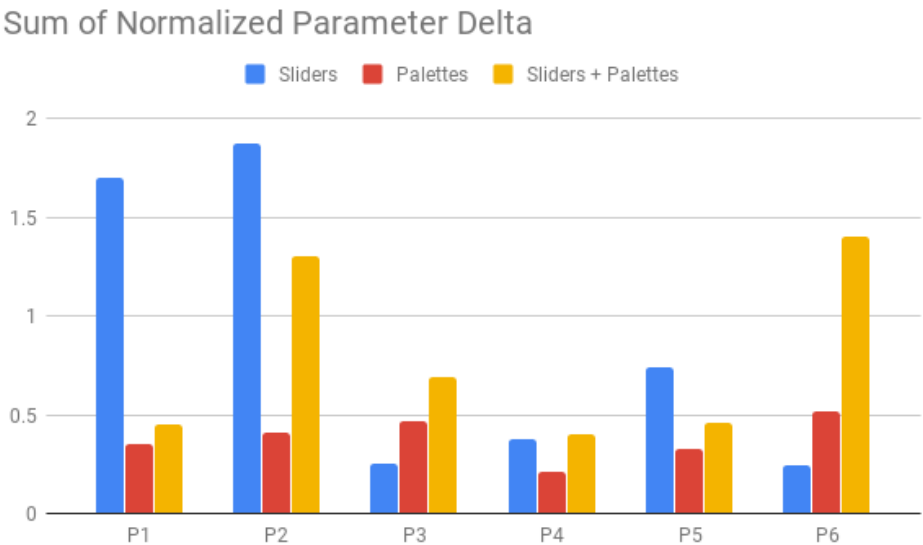


Figure 5.13: Task I Distance to the Target

5.3.2 Task II: Free Exploration

5.3.2.1 Completion Time (only for reference)

In the experiment, I told the participants that the requirements for Task II is merely to reach an effect they really like. The completion time of the task is only for reference.

Table 5.4 and Figure 5.14 shows the completion time (in seconds) for each participant.

UI Condition	P1	P2	P3	P4	P5	P6
Sliders	148	136	85	247	294	190
Palettes	239	179	111	281	124	138
Both	194	268	146	308	210	169

Table 5.4: Task II Completion Time (seconds)

Again, least completion time does not necessarily mean the UI condition works most efficiently, and longest completion time is not to say the corresponding UI condition forges a most immersive state of flow for the participants, as different participants have different backgrounds and preferences,

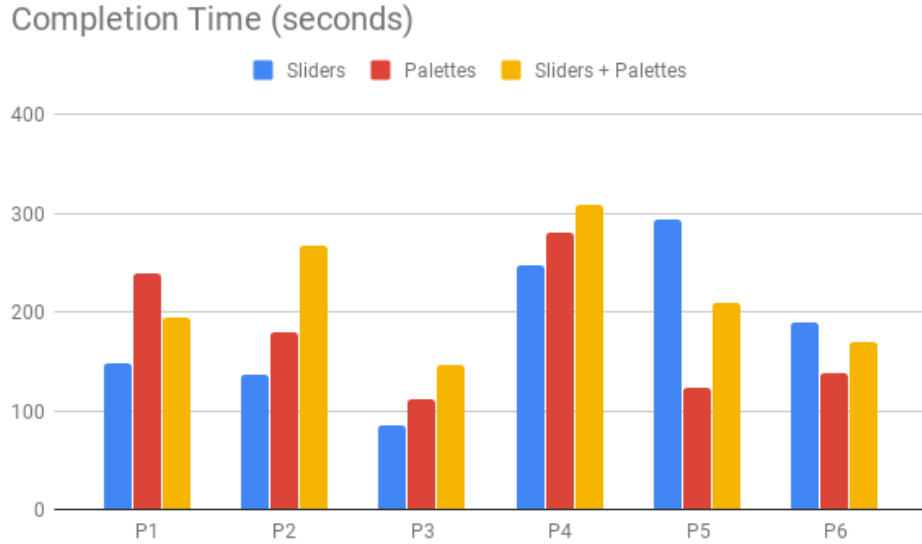


Figure 5.14: Task II Completion Time (seconds)

learn new things at different paces, and take different attitudes towards this task.

But we can still see from Table 5.4 that for the majority of the participants (4/6), if the UI condition comprises palettes, they spent more time in the free exploration process. This is an indication that people like to play with the Palettes more than the Sliders. However, it may also mean that Palettes are not accurate enough for the participants to reach what they want. To clarify the reasons behind, I need further analysis on the qualitative results.

## 5.4 Subjective Results

Subjective results also come from the data of the same 36 experimental trials collected ( $6 \text{ PARTICIPANT} \times 2 \text{ TASK} \times 3 \text{ UI CONDITION} \times 1 \text{ TRIAL}$ ).

### 5.4.1 Self-Reported Measures (Quantitative)

After completing each small task, participants were asked to rate six statements on a 5-point likert scale, from strongly disagree to strongly agree. The statements asked whether the current UI condition in the small task helped

them to reach the target effect: accurately, quickly, easily, confidently, comfortably, and enjoyably.

Then after completing each big task (consisting of 3 small tasks with the 3 UI conditions), participants were asked to compare the 3 UI conditions (Sliders, Palettes, Sliders + Palettes) for achieving the same goal: rank for each of the UI conditions on a 5-point likert scale, from "not like at all" to "like very much", thus state their preferences.

5.4.1.1 Task I: Reaching a Target Effect

Table 5.5 and Figure 5.15 shows the average grading from all participants.

UI Condition	Accurately	Quickly	Easily	Confidently	Comfortably	Enjoyably
Sliders	3.33	3	3.17	2.67	3.67	3.67
Palettes	2.67	3.67	3.67	3	3.67	4.17
Both	3.83	4.17	3.5	3.5	4	4.17

Table 5.5: Task I Self-Reported Average Grading

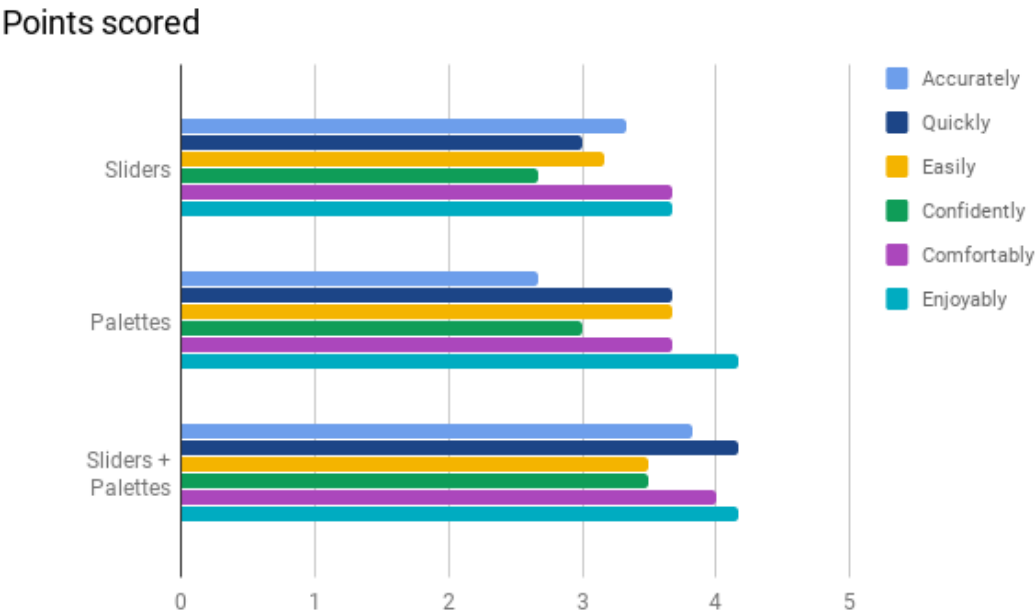


Figure 5.15: Task I Self-Reported Average Grading

Table 5.6 and Figure 5.16 shows the sum overall ranking for all participants.

UI Condition	Not like at all	Not like much	Neutral	Kind of like	Like very much
Sliders	0	0	2	4	0
Palettes	0	0	1	3	2
Both	1	0	0	1	4

Table 5.6: Task I Self-Reported Overall Ranking

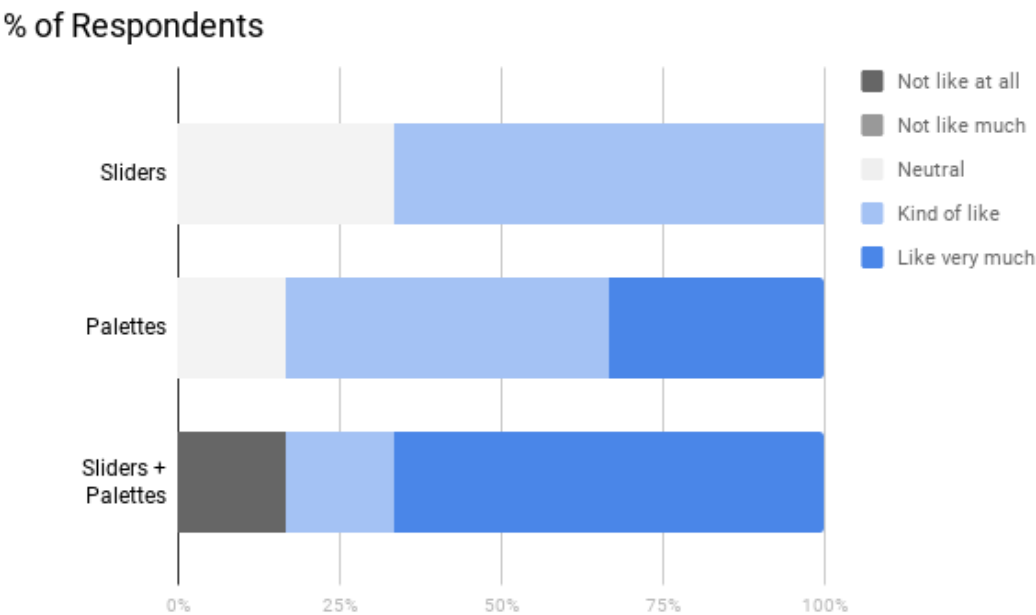


Figure 5.16: Task I Self-Reported Overall Ranking

From the average grading and overall ranking for Task I, we can see that the UI condition with both the Sliders and the Palettes is the most popular. For task I, participants felt the Palettes can help them finish the task more quickly, more easily, more confidently and more enjoyably. They felt the Sliders are more accurate.

**5.4.1.2 Task II: Free Exploration**

Table 5.7 and Figure 5.17 shows the average grading from all participants.

UI Condition	Accurately	Quickly	Easily	Confidently	Comfortably	Enjoyably
Sliders	4.33	3.83	3.67	3.83	3.83	4
Palettes	3.5	3.83	3.75	3.67	3.67	4.33
Both	4.33	4.25	4.17	4	4.67	4.83

Table 5.7: Task II Self-Reported Average Grading

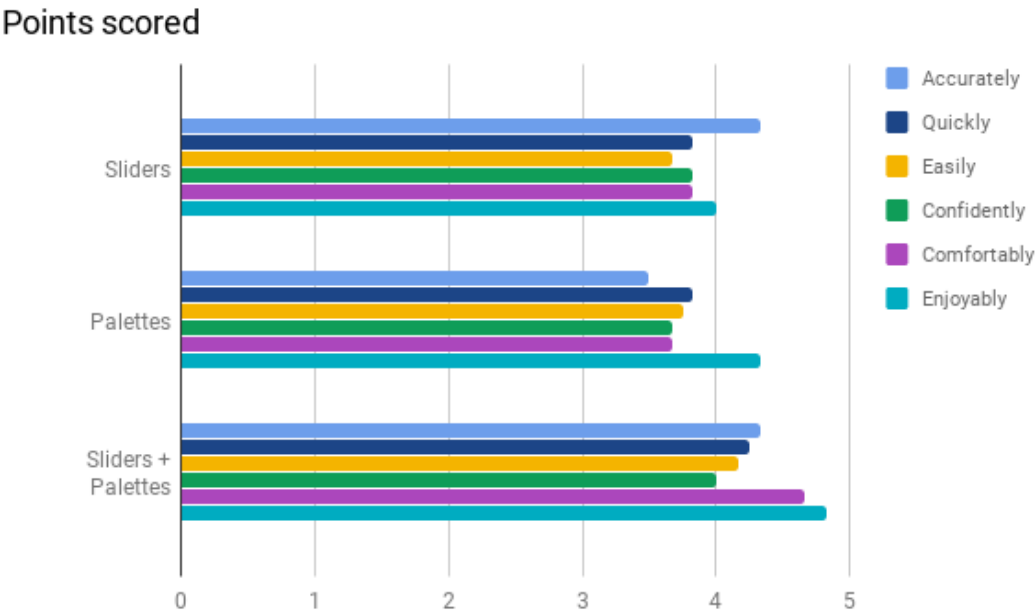


Figure 5.17: Task II Self-Reported Average Grading

Table 5.8 and Figure 5.18 shows the sum overall ranking for all participants.

From the average grading and overall ranking for Task II, we can see that the UI condition with both the Sliders and the Palettes is again the most popular. For task II, participants felt the Palettes can help them finish the task more easily and more enjoyably, but less confidently and less comfortably. This time, they still felt the Sliders more accurate, but they did not feel the Palettes quicker.

UI Condition	Not like at all	Not like much	Neutral	Kind of like	Like very much
Sliders	0	0	1	5	0
Palettes	0	2	1	2	1
Both	0	0	0	2	4

Table 5.8: Task II Self-Reported Overall Ranking

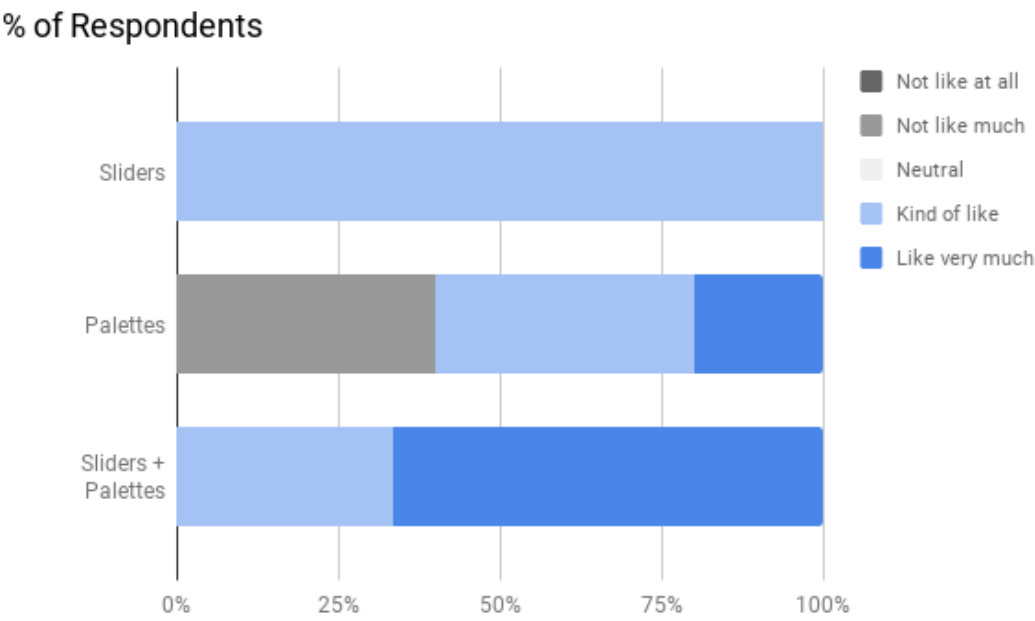


Figure 5.18: Task II Self-Reported Overall Ranking

## 5.4.2 Debriefing on Strategies and Preferences (Qualitative)

### 5.4.2.1 Task I: Reaching a Target Effect

All participants (6/6) tried to extract features from the target effect. For example, some of them described Target A as: the visual effect rotates anti-clockwise on the whole, but for each particle, the moving direction is a bit messy, and each of them has a small tail. In their trial and error process, they tried to reach each of the extracted features until they thought the effect state was close enough to the target effect.

Most participants were most satisfied with the UI condition in which both Sliders and Palettes were in permission to use. In that UI condition, participants showed 3 typical mindsets or behavior patterns: adopting Palettes willingly and relying heavily on Palettes, combining Palettes and Sliders, and adopting Palettes reluctantly.

#### Adopting Palettes Willingly

Some participants (2/6) hardly used the Sliders at all. P1 said, "It is hard to remember all the parameter meanings or what they control. Even if I test them out in the beginning, soon I forget most of them during the task." Even for the small task with the UI condition of Sliders only, she used the Sliders to get 3 slightly different states close to the target effect, and created a Palette with the 3 states to try if she could reach the target effect.

When she operated on the Palettes, she took the strategy of checking out all the extreme states first. She said it was lucky for her to find an extreme state close enough to the target effect, and test around in the neighboring area. Asked about how she felt using the Palettes, she answered, "I like the palettes, because it's like a game. It's interactive and once you know how to do it, it's intuitive."

P5 was not so tactical. After fumbling with the Sliders and giving them up, she tried the Palettes for some time. Finally she found out the closest extreme state and searched nearby.

#### Combining Palettes and Sliders

Some participants (2/6) took the strategy of finding a globally approximate state with the Palettes and then adjusting parameters with Sliders to diminish small differences.

After P2 and P3 reached an approximate state with the Palettes, they compared it to the target effect, and extract features that did not match.

Then they tried the Sliders to understand what each parameter controls, and adjusted the right ones.

When testing out Sliders, P2 and P3 took the strategy of going over all the Sliders from minimum value to maximum value, and trying to figure out what each parameter controls. P3 took a step further as he had to be sure about what he had changed, once he found it was not the right parameter, he would change back to its previous value.

Both of them felt the Palettes are a quick tool to find an approximate state, but lack precision. The Sliders can achieve more accurate results, but it takes a long time to learn.

### **Adopting Palettes Reluctantly**

Some participants (2/6) relied heavily on the Sliders.

P4 had a difficult time adopting the new UI, i.e. the Palettes. She took longer time with the UI condition of the Palette than other participants. She explained, "I have a very scientific mindset. I need to understand the system, which means how each parameter affects the visualization." When she used the Palettes, unlike other participants moving inside the Palettes freely and trying out as many interpolation states as possible (especially the extreme states and states on the edges of the Palettes), she hesitated and staggered as she was not sure where to move. However, in the end she successfully completed the small task.

In the small task with the UI condition of the Sliders, she was very careful as she tested a single variable at a time, much like P3's strategy, and she successfully identified the 3 parameters that needed to be changed. In the small task with the UI condition of both the Palettes and the Sliders, she also took a similar strategy as P3 to combine both to finish the task, but she soon lost interest in adjusting the Sliders.

P6 was very quick to learn the usage of the Palettes. But in Task I with both the Sliders and the Palettes, he soon deleted the Palettes and relied only on the Sliders. He was also an exception to rate the overall ranking for the UI condition with both the Sliders and the Palettes as "not like at all". When asked why, he answered, "To reach a target effect, I can only focus on 1 UI to figure out the distance. I cannot look at both at the same time."

#### **5.4.2.2 Task II: Free Exploration**

In the free exploration task, again most participants were most satisfied with the UI condition in which both Sliders and Palettes were in permission to use. But in Task II they like Sliders more than in Task I. As P1 said, "It



was fine this time, because I had to explore different visual effects, so I could stop whenever I wanted. But for the first task, when I had to reach a specific target, this didn't work well, as it was more difficult to manage."

In this task, participants also showed 3 mindsets or behavior patterns: welcoming surprises, combining surprises and precision, and requiring absolute control.

### **Welcoming Surprises**

Some participants (2/6) did not care about precision that much. When they could freely choose between the Sliders and the Palettes, they relied heavily on the Palettes.

P1 tried out 9 presets and adjusted some of them with the Sliders and saved the ones she likes. After that she never touched the Sliders again. She created a Palette with 6 saved states, explored and saved another 2 interpolation states. Then she chose the 2 newly saved states with 2 previously saved ones to create another Palette. After some time of exploration she settled on a final interpolation state. She said, "The palette is really fun. I like surprises when I explore it."

When I asked P1 why she did not use the Sliders that much, she told me, "I only used them at the beginning, because I prefer the Palettes, as they are more enjoyable and fun to use. Besides, I'm still confused about the immediate correlation between the parameter values and the visual effect. I cannot direct the visual effect to the way I want with the Sliders."

P6 took a very similar strategy as P1, but what's different is that he knows how the parameters control the visual effect. According to him, he did not have a goal to reach in his mind, so he created some effects of his own with features that are almost opposite extremes, and started exploration in the Palette he built with those effects. He said, "I was just trying and would like to see how it goes, and soon enough I get something I like."

### **Combining Surprises and Precision**

Half of the participants (3/6) want the best of both worlds, meaning they would like to combine surprises and precision.

In the small task with the UI condition of only the Palettes, P2 created a Palette with 3 presets and saved 4 interpolation states. But she was not contented with the results and requested for Sliders. Then she chose one of the saved interpolation state and modified 4 parameters until she reached a satisfying result.

P3 and P5 took the same strategy with P2 when they could manipulate both the Sliders and the Palettes. But P3 soon lost interest in modifying the parameters one by one and gave up further adjustments. P5 was quick to modify the right parameter slider and get to an effect she likes.

Before the final touch with the Sliders, P5 explored with the Palettes. She created one palette with 5 extreme states, and placed 3 preferred extreme states in the center, 2 others far away, as she wanted the final effect to be more influenced by the ones she likes better.

### **Requiring Precise Control**

One participant (1/6) did not like surprises at all. She wanted to understand the system and obtain control over it. When she could freely choose between the Sliders and the Palettes, she relied heavily on the Sliders.

Given the 9 presets, P4 tried to compare the parameter values and features of the presets, to gain understanding of the system. She did not really use the Palettes when both are allowed — she tried the Palettes but soon aborted. According to her, she found some features in the presets appealing and wanted to combine those features, but the Palettes did not help in that. In the end, she chose a preset close to what she likes, and tuned from there. She would think about certain features in her mind, and find parameters that control those features. She was satisfied about precise control over the final result.

## **5.5 Discussion**

According to participants' feedback, the design of Motion Palette can be improved in many ways. In summary I find the following four design implications.

### **5.5.1 New Interpolation Strategy**

#### **Non-linear Interpolation**

Apart from linear interpolation in palettes, gradient ramp feature can be added to allow non-linear interpolation.

#### **Blending Visual Features**

P4 and P6 found out the Palettes cannot help in combining different appealing features together from saved states, but can only calculate the interpola-

tion values of the parameters. They suggested that it would be much better if the Palettes could extract features from the extreme states and blend these features.

### 5.5.2 Natural Interaction with Palettes

#### Adding/Deleting a State to/from a Palette

P1 and P5 want to drag a state to a palette, so that the palette can add the dragged state as a new extreme state. P3 wants to delete an extreme state from a palette.

#### Trigger to switch modes for exploring/recording

P2 thinks she needs a trigger, e.g. dragging, to activate exploration inside palettes, so that the state will not change when she deactivates exploration mode and moves the mouse cursor inside the palette.

### 5.5.3 Bidirectional Control between Palettes and Sliders

Currently, palettes can affect sliders, but sliders can not affect back palettes. P2 thinks it is a problem and affects how she can play with both the Sliders and the Palettes at the same time, "I didn't like it when I switch to the Palettes, I could lose the parameter values in the Sliders." That is mainly due to the reason that the Palettes represent a narrower parametric space than the Sliders. Once figuring out the mapping in-between, the control between Sliders and Palettes can be made bidirectional.

### 5.5.4 Machine Learning User Preferences

P3 thinks it is trivial to find extreme states with the Sliders. P3 and P6 think the 9 presets in the free exploration task limit the possibilities of the visual effects that can be created. Further study can combine reinforcement learning to help users find extreme states.

## Chapter 6

# Conclusions

### 6.1 Conclusion

In this thesis, my major contribution comes in three aspects. Firstly, I uncovered exploratory needs of parametric designers in current motion design tools. Secondly, I designed Motion Palette, a new solution that enables customized motion exploration for visual artists specialized in dynamic visual effects design. Thirdly, I conducted an experiment to systematically observe how designers interact with Motion Palette, in order to improve this new tool.

#### 6.1.1 Uncover Exploratory Needs for Motion Designers

In Study One, I interviewed 10 designers and artists about their creative process for their recent projects involving parametric design and generative art. From the interviews I found 5 behavior patterns and related breakdowns and workarounds in their work process. Then I identified design implications and opportunities to support their creative work.

#### 6.1.2 Design and Evaluate Motion Palette

In Study Two, I designed a new creativity support tool, Motion Palette, to facilitate motion exploration for artists specialized in dynamic visual effects design. Experiments with potential users show that the combination of this new tool with the traditional parametric exploration tool e.g. the sliders, work best in both the task of reaching a target effect and free exploration to create the users' preferred effects.

## 6.2 Future Work

Future studies can extend the Motion Palette in many ways. For example, combine reinforcement learning to find extreme states for creating palettes, apply to shader art, allow designers and artists to define their own customized upper level motion space, etc.

The same notion of Motion Palette can also be applied to many more scenarios beyond dynamic visual effects design. For example, explore the shape of buildings in architecture design, tune character animation, design a continuous transitioning parametric space for controlled experiments, etc.

# Bibliography

- [1] ATTAR, R., AISH, R., STAM, J., BRINSMEAD, D., TESSIER, A., GLUECK, M., AND KHAN, A. Physics-based generative design. In *CAAD Futures Foundation* (2009).
- [2] AUTODESK. Slate material editor. <https://www.youtube.com/watch?v=CLar6qhZKUY>, 2010. Accessed: 2019-08-18.
- [3] AUTODESK. Base, target, and blend shapes. <https://knowledge.autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2018/ENU/Maya-CharacterAnimation/files/GUID-01EFEC6D-41EA-46AA-81B1-C171DA4316F4-htm.html>, 2018. Accessed: 2019-08-18.
- [4] BARTIER, P. M., AND KELLER, C. Multivariate interpolation to incorporate thematic surface data using inverse distance weighting (idw). *Computers & Geosciences* 22, 7 (1996), 795 – 799.
- [5] BEAUDOUIN-LAFON, M. Instrumental interaction: An interaction model for designing post-wimp user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2000), CHI '00, ACM, pp. 446–453.
- [6] BEAUDOUIN-LAFON, M. Towards unified principles of interaction. In *Proceedings of the 12th Biannual Conference on Italian SIGCHI Chapter* (New York, NY, USA, 2017), CHIItaly '17, ACM, pp. 1:1–1:2.
- [7] BEAUDOUIN-LAFON, M., AND MACKAY, W. E. Reification, polymorphism and reuse: Three principles for designing visual interfaces. In *Proceedings of the Working Conference on Advanced Visual Interfaces* (New York, NY, USA, 2000), AVI '00, ACM, pp. 102–109.
- [8] BRAUN, V., CLARKE, V., HAYFIELD, N., AND TERRY, G. *Thematic Analysis*. Springer Singapore, Singapore, 2019, pp. 843–860.

- [9] BRUCKNER, S., AND MOLLER, T. Result-driven exploration of simulation parameter spaces for visual effects design. *IEEE Transactions on Visualization and Computer Graphics* 16, 6 (Nov 2010), 1468–1476.
- [10] CODEPEN. Codepen.io. <https://codepen.io/>, 2012. Accessed: 2019-08-18.
- [11] EISELMAYER, A., WACHARAMANOTHAM, C., BEAUDOUIN-LAFON, M., AND MACKAY, W. E. Touchstone2: An interactive environment for exploring trade-offs in hci experiment design. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2019), CHI '19, ACM, pp. 217:1–217:11.
- [12] ET AL., F. I. P. *Computer Facial Animation*. CRC Press, 2008, ch. 3, pp. 43–84.
- [13] FIGMA. Figma. <https://www.figma.com/>, 2016. Accessed: 2019-08-18.
- [14] GARCIA, J., TSANDILAS, T., AGON, C., AND MACKAY, W. E. Structured observation with polyphony: A multifaceted tool for studying music composition. In *Proceedings of the 2014 Conference on Designing Interactive Systems* (New York, NY, USA, 2014), DIS '14, ACM, pp. 199–208.
- [15] JACOBS, J., GOGIA, S., MĚCH, R., AND BRANDT, J. R. Supporting expressive procedural art creation through direct manipulation. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2017), CHI '17, ACM, pp. 6330–6341.
- [16] KAZI, R. H., CHEVALIER, F., GROSSMAN, T., AND FITZMAURICE, G. Kitty: Sketching dynamic and interactive illustrations. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, 2014), UIST '14, ACM, pp. 395–405.
- [17] KAZI, R. H., GROSSMAN, T., CHEONG, H., HASHEMI, A., AND FITZMAURICE, G. Dreamsketch: Early stage 3d design explorations with sketching and generative design. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, 2017), UIST '17, ACM, pp. 401–414.
- [18] KWAN-LIU MA. Image graphs-a novel approach to visual data exploration. In *Proceedings Visualization '99 (Cat. No.99CB37067)* (Oct 1999), pp. 81–88.

- [19] MACKAY, W. *Video to Support Interaction Design, DVD*. ACM, New York, 2002.
- [20] MARKS, J., ANDALMAN, B., BEARDSLEY, P. A., FREEMAN, W., GIBSON, S., HODGINS, J., KANG, T., MIRTICH, B., PFISTER, H., RUMML, W., RYALL, K., SEIMS, J., AND SHIEBER, S. Design galleries: A general approach to setting parameters for computer graphics and animation. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1997), SIGGRAPH '97, ACM Press/Addison-Wesley Publishing Co., pp. 389–400.
- [21] MAUDET, N., JALAL, G., TCHERNAVSKIJ, P., BEAUDOUIN-LAFON, M., AND MACKAY, W. E. Beyond grids: Interactive graphical substrates to structure digital layout. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2017), CHI '17, ACM, pp. 5053–5064.
- [22] MÜLLER, P., MÜLLER ARISONA, S., SCHUBIGER-BANZ, S., AND SPECHT, M. Interactive editing of live visuals. In *Advances in Computer Graphics and Computer Vision* (Berlin, Heidelberg, 2007), J. Braz, A. Ranchordas, H. Araújo, and J. Jorge, Eds., Springer Berlin Heidelberg, pp. 169–184.
- [23] OPENFRAMEWORKS. openframeworks. <https://openframeworks.cc/>, 2005. Accessed: 2019-08-18.
- [24] PEREVALOV, D. *Mastering openFrameworks: Creative Coding Demystified*. Packt Publishing Limited, 2013, ch. 3, pp. 59–82.
- [25] SHUGRINA, M., LU, J., AND DIVERDI, S. Playful palette: An interactive parametric color mixer for artists. *ACM Trans. Graph.* 36, 4 (July 2017), 61:1–61:10.
- [26] SMITH, R., PAWLICKI, R., KÓKAI, I., FINGER, J., AND VETTER, T. Navigating in a shape space of registered models. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (Nov 2007), 1552–1559.
- [27] UNITY. Shader graph. <https://unity.com/shader-graph>, 2017. Accessed: 2019-08-18.
- [28] VICTOR, B. Dynamic pictures. <http://worrydream.com/DynamicPicturesMotivation>, 2011. Accessed: 2019-08-18.



- [29] VICTOR, B. Up and down the ladder of abstraction. <http://worrydream.com/LadderOfAbstraction/>, 2011. Accessed: 2019-08-18.
- [30] VICTOR, B. Inventing on principle. In *Proc. of the Canadian University Software Engineering Conference* (2012).
- [31] VICTOR, B. Stop drawing dead fish. In *ACM SIGGRAPH 2012 Talks* (2012), SIGGRAPH '12.
- [32] XIA, H., ARAUJO, B., GROSSMAN, T., AND WIGDOR, D. Object-oriented drawing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2016), CHI '16, ACM, pp. 4610–4621.
- [33] XING, J., KAZI, R. H., GROSSMAN, T., WEI, L.-Y., STAM, J., AND FITZMAURICE, G. Energy-brushes: Interactive tools for illustrating stylized elemental dynamics. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (New York, NY, USA, 2016), UIST '16, ACM, pp. 755–766.
- [34] ZSOLNAI-FEHÉR, K., WONKA, P., AND WIMMER, M. Gaussian material synthesis. *ACM Trans. Graph.* 37, 4 (July 2018), 76:1–76:14.